

One Tile to Rule Them All: Simulating Any Turing Machine, Tile Assembly System, or Tiling System with a Single Puzzle Piece

<p>Erik D. Demaine[*] CSAIL MIT Cambridge, MA 02139, USA edemaine@mit.edu</p>	<p>Martin L. Demaine[*] CSAIL MIT Cambridge, MA 02139, USA mdemaine@mit.edu</p>	<p>Sándor P. Fekete Computer Science TU Braunschweig Germany s.fekete@tu-bs.de</p>
<p>Matthew J. Patitz[†] Dept. of CSCE University of Arkansas Fayetteville, AR 72701, USA patitz@uark.edu</p>	<p>Robert T. Schweller[†] Computer Science University of Texas–Pan American Edinburg, TX 78539, USA rtschweller@utpa.edu</p>	
<p>Andrew Winslow[‡] Computer Science Tufts University Medford, MA 02155, USA. awinslow@cs.tufts.edu</p>	<p>Damien Woods[§] Computer Science California Institute of Technology Pasadena, CA 91125, USA woods@caltech.edu</p>	

Keywords: Self-assembly, tiling, aperiodic tile set, universality, one

Abstract

In this paper we explore the power of tile self-assembly models that extend the well-studied abstract Tile Assembly Model (aTAM) by permitting tiles of shapes beyond unit squares. Our main result shows the surprising fact that any aTAM system, consisting of many different tile types, can be simulated by a *single* tile type of a general shape. As a consequence, we obtain a single *universal* tile type of a single (constant-size) shape that serves as a “universal tile machine”: the single universal tile type can simulate any desired aTAM system when given a single seed assembly that encodes the desired aTAM system. We also show how to adapt this result to convert any of a variety of plane tiling systems (such as Wang tiles) into a “nearly” plane tiling system with a single tile (but with small gaps between the tiles). All of these results rely on the ability to both rotate and translate tiles; by contrast, we show that a single nonrotatable tile, of arbitrary shape, can produce assemblies which either grow infinitely or cannot grow at all, implying drastically limited computational power. On the positive side, we

^{*}Research supported in part by NSF grant CDI-0941538.

[†]Research supported in part by NSF grant CCF-1117672.

[‡]Research supported in part by NSF grant CDI-0941538.

[§]Supported by NSF grants 0832824 (the Molecular Programming Project), CCF-1219274, and CCF-1162589.

show how to simulate arbitrary cellular automata for a limited number of steps using a single nonrotatable tile and a linear-size seed assembly.

1 Introduction

This paper shows that a single rotatable tile type suffices to simulate any desired tile system (consisting of many different tile types) in two different worlds: classic geometric tiling and self-assembling tile systems.

Classic geometric tiling Tiling the plane with geometric shapes goes back to Kepler in 1619; refer to the classic survey book [9]. Wang [22] introduced the geometrically simpler model of tiling the plane with (fixed-orientation) unit-square tiles with colored edges such that abutting edges match in color. Such edge-matching tiles can be converted to purely geometric tiles, using jigsaw-like tabs and pockets to simulate colors; a reverse translation is possible by losing some factors in scale and number of tiles [5].

The tiling literature often aims for minimal tile sets to achieve certain properties. For example, Wang’s student Berger [1] first found an aperiodic set of 20,426 Wang tiles, and a sequence of improvements by Berger, Knuth, Läuchli, Robinson, Penrose, Ammann, and Culik finally resulted in an aperiodic set of just 13 Wang tiles [4]. Using geometric tiles (or Wang tiles that can rotate and reflect, and a complementary matching condition), Robinson [17] obtained just 6 tiles, while the famous Penrose tilings [15] achieve aperiodicity with just two tiles. It remains open whether one tile suffices to obtain aperiodicity; such a result has been obtained given nonlocal matching rules among tiles [21], or with overlaps [10].

Tilings are closely related to computation. Berger [2] proved that it is undecidable to determine whether a set of Wang tiles (with infinitely many copies of each) tile the plane, by simulating a Turing machine; Robinson [17] gave a simpler proof. Both proofs use an unbounded number of Wang tiles, growing with the size of the Turing machine.

Our results I: Universal tiling with one tile We prove in Section 7 that these and many other plane tiling systems, consisting of any (finite) number of tiles, can be simulated by a “nearly plane” tiling that uses only a single tile. The input tiling system can live on either a square or hexagonal grid, can allow tiles to rotate or not, can allow tiles to reflect or not, and can define compatible tile adjacencies according to matching color (as in Wang tiling) or complementary color pairs (as in actual DNA). The output one-tile system requires tiles to live on the same square or hexagonal lattice, allows tiles to rotate, and is *nearly plane* tiling in the sense that it leaves tiny gaps between the tiles.

Our simulation result applies in particular to all Wang tilings, even allowing rotation as in Robinson’s tilings, which has several important implications:

1. We obtain for the first time (in any model of tiling) that a single tile can simulate any desired Turing computation. (Here the combinatorial complexity of the tile, instead of the number of tiles, depends on the size of the Turing machine.)
2. We obtain for the first time (in any model of tiling with a local matching rule and no overlaps) that a single tile suffices to produce aperiodic (nearly plane) tilings.

3. We obtain for the first time (in any model of tiling) that a single tile can simulate a countably infinite number of tiling systems (a form of intrinsic universality).

Before proving these results, we develop a number of techniques for another model: tile self-assembly. In fact, the above results are essentially achieved as corollaries, with some additional technical detail, of our tile self-assembly results, the proofs of the latter being somewhat more involved and constitute the main portion of the paper.

Tile self-assembly Winfree [23] introduced the *abstract Tile Assembly Model* (aTAM) as a clean theoretical model that approximates the reality of nanoscale self-assembling systems. In several experiments of increasing complexity and reliability (e.g., [20, 19]), this model has been shown to be physically practical, with tiles made up from DNA strands. As a result, the model has become the standard in theoretical work on self-assembly.

The aTAM is essentially a more local (and thus more realistic) version of Wang tiling: we start with a specific tile (called the *seed*), and repeatedly add any tile to the assembly that has enough matching glues (colored edges) to “stick” to the rest of the assembly. Here we assign an integer *strength* to each glue type (color class), which represents the affinity/attraction for matching glues of that type, and specify a global *temperature* (typically 2) specifying the total required strength for a tile to attach to the assembly. Unlike Wang tiling, in the aTAM we can never throw away partially formed assemblies and, in fact, the aTAM can be thought of as a special kind of asynchronous, and nondeterministic, cellular automaton. See Section 2 for details.

Our results II: Universal self-assembly with one tile We prove in Section 5 that any aTAM system—consisting of any (finite) number of tiles—can be simulated by just a single tile, in a generalization of the aTAM model called pfbTAM (polygonal free-body Tile Assembly Model). Precisely, we show that any temperature- τ aTAM system can be converted into a temperature- τ pfbTAM system of a single tile type such that the two systems have exactly the same producible assemblies (modulo isometry). This construction is *self-seeding* in the sense that it starts from a single copy of the very same tile; it is even a challenge to get the next copy of the tile to attach without allowing the construction to grow indefinitely.

Combining this result with intrinsic universality of aTAM [6], we construct a single constant-size tile t_U such that any temperature- τ aTAM system Γ can be converted into a seed assembly such that adding (infinitely many copies of) tile t_U to the seed in the temperature-2 pfbTAM simulates Γ . This tile t_U is a kind of geometric analog to a universal Turing machine, simultaneously simulating the shape construction and computational ability of an arbitrary tile assembly system. (On the other hand, no standard universal Turing machine has only one state, or only one symbol [14].)

The pfbTAM model differs from aTAM in two ways—allowing polygonal tiles instead of just squares and allowing tiles to rotate—both of which we claim are physically realistic. For example, DNA origami [18] is a rapidly evolving technology that has been used to successfully build numerous complex shapes using strands of DNA. The technology has evolved to the point where free software automatically designs DNA to fold into essentially arbitrary desired shapes. aTAM-like tiles with polyomino (instead of square) shapes have already been developed in practice [24] and studied in theory [8]. Second, rotation is a natural addition, as nanoscale objects do rotate in reality; the aTAM omitted rotation, as it was easier to work with translation only, and with only square tiles, it turns out to be equivalent.

Given that our tile rotates and translates, a natural question is what can be achieved with *translation only*? We show that allowing rotation is essential: the same type of results are impossible with translation-only polygonal tiles. Specifically, we prove in Section 6.2 that any single-tile translation-only pfbTAM self-assembly system either can grow infinitely (producing assemblies of unbounded size) or cannot grow at all (producing just the seed tile). This negative result holds even if the system starts from a more complex seed assembly consisting of up to three copies of the tile, even though such a seed avoids the difficult issue of getting the first bond between two tiles. On the positive side, though, in Section 6.1 we show that a seed assembly consisting of $O(k)$ copies of the single tile suffices to simulate an arbitrary 1D cellular automaton for k steps. In general, we conjecture that no finite seed suffices for unbounded computational power with translation only, in stark contrast to allowing rotation.

Our results III: Hexagonal tile assembly systems Along the way, we consider in Section 3 aTAM systems with unit-sized hexagonal tiles on a hexagonal grid. The only previous paper considering this model [11] simply showed differences between squares and hexagons with respect to infinite constructions. Here we show that any temperature-2 square aTAM system can be simulated by a temperature-2 hexagonal aTAM system in which all glues have strength at most 1. The construction works at a scale factor of only 3: each square tile is simulated by a 3×3 block of hexagonal tiles. By contrast, any temperature-2 square aTAM system in which all glues have strength at most 1 can not grow outside its bounding box; we use the fact that hexagons have the ability to continue to grow.

This result is a key step to proving our main positive result (aTAM simulation allowing translation and rotation). Specifically, we show in Section 5 how to simulate any temperature-2 hexagonal aTAM system, that uses strength-1 glues, by a rotatable polygon that mimics different tile types by attaching at different rotations. More generally, we believe that our result on hexagonal aTAM may have independent importance in the study of self-assembly systems.

2 Models

2.1 The Polygonal Free-Body Tile Assembly Model

The *polygonal free-body Tile Assembly Model* (or *pfbTAM*) generalizes self-assembly models such as the aTAM by using arbitrary geometric shapes for the constituent parts that can be translated and rotated by any amount. In our positive results, we focus on convex regular n -gons with small surface geometries; our negative results are valid for arbitrary shapes, as discussed in Section 6.2.

The basic units of the pfbTAM are polygonal tiles, and a pfbTAM system Γ is defined as $\Gamma = (F, \tau, \sigma)$, where F is a finite set of polygonal tile types, $\tau \in \mathbb{N}$ is the *temperature* parameter for the system, and σ is a *seed assembly* consisting of a set of polygonal tiles from F and their locations.

Tiles A *polygonal tile* in the pfbTAM model is bounded by a simple closed polygon enclosing the tile’s interior. Boundaries of two tiles may intersect, but not their interiors.

Glues Let Σ be an alphabet of *glue types*. Each glue type $g \in \Sigma$ is assigned a value *strength* ($g \in \mathbb{N}$). The boundary of the polygon tile is divided into intervals called *sides* (which may be more

complex than single line segments), and each interval is assigned a glue type from Σ . A pair of boundary intervals are *coincident* if there is a bijection between pairs of points on the two intervals such that each pair is coincident. For a given pair of sides on distinct polygon tiles, we define three types of geometric compatibility, based upon whether the tiles can be oriented such that portions of their boundaries are coincident. See Figure 1 for a description of geometric compatibility.

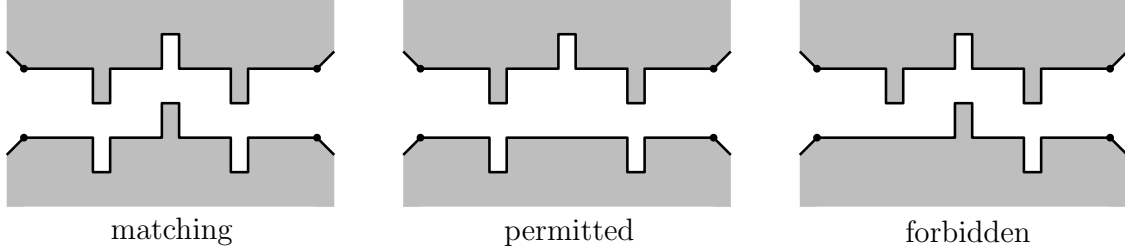


Figure 1: The three possible types of side-side interactions between a pair of polygon free-body tiles. Matching sides form a bond with strength according to their glues, while permitted and forbidden sides do not form a bond.

If there exists a pair of tile orientations so that the pair of sides become fully coincident, then these sides have *matching geometry*. If a pair of sides do not have matching geometry, but can be oriented such that the endpoints (and possibly more) of the sides are coincident, then the sides have *permitted geometry*. Finally, if a pair of sides do not have permitted geometry, then they have *forbidden geometry*.

Bonding between tiles occurs via the interaction of side pairs. If a pair of sides have permitted or forbidden geometry, then they cannot form a bond, regardless of glues. If the sides instead have matching geometry, then they can bond with strength according to their glues. Matching glues form a bond with strength determined by the glue type. Non-matching glues form zero-strength bonds.

Systems and assembly An *assembly* is a collection of tiles that are bound together by their adjacent glues; an assembly is τ -*stable* if any possible way of separating it into two disconnected assemblies must break glue bonds that sum to at least the value τ . Formally, we define the *bond graph* of an assembly to be the planar graph consisting of a labeled node, and an edge between two tiles iff they have edges that form a bond. A tile can attach to an assembly if it is able to bind with strength at least τ .

Assembly of a pfbTAM system $\Gamma = (F, \tau, \sigma)$ starts with the seed assembly σ and proceeds as individual tiles from F nondeterministically bind τ -stably and one at a time to the growing assembly. An assembly is *producible* if there is a sequence of τ -stable assemblies that produces it; a producible assembly is *terminal* if there are no tiles that can τ -stably attach to it. The set of producible assemblies of Γ is $PROD_\Gamma$, and the set of terminal assemblies is $TERM_\Gamma$. A Γ is *directed* (a.k.a. *deterministic*, *confluent*) if there is a single producible, terminal assembly (i.e. $|TERM_\Gamma| = 1$).

Rotationally restricted systems In some cases we restrict the granularity in which polygonal tiles may rotate, and whether or not they can flip. A ϕ -restricted system limits $PROD_\Gamma$ to only contains assemblies that are obtained through attachments of tiles in F that are translations and

flips and rotations of degrees $c \cdot \phi$ for integers c . Rotationally restricted systems with $\phi = 0$ that do not allow flipping are referred to as *translation-only* systems.

2.2 aTAM and hTAM

aTAM The abstract Tile Assembly Model can be formulated as a restricted version of the pfbTAM. An aTAM system is a translation-only pfbTAM system $\Gamma = (T, \tau, s)$ such that T consists of equal size squares in the same rotational orientation with one glue type assigned to each tile side.

hTAM The hexagonal Tile Assembly Model is a restricted version of the pfbTAM. An hTAM system is a translation-only pfbTAM system $\Gamma = (T, \tau, s)$ such that T consists of equal size regular hexagons in the same rotational orientation with one glue type assigned to each tile side.

2.3 Simulation

In this section we define what it means for the pfbTAM to simulate the hTAM and aTAM and for the hTAM to simulate the aTAM.

Simulating hTAM systems with pfbTAM systems At a high level, we say a pfbTAM system $\Gamma_f = (T', \tau', \sigma)$ simulates an hTAM system $\Gamma_h = (T, \tau, \sigma)$ if there is a bijection between orientations (composed of rotations and flips) of tiles in T' and tiles in T such that this bijection yields a second bijection between producible assemblies of Γ_f and Γ_h , where both tile location and bond structure is preserved.

A pfbTAM system $\Gamma_f = (T', \tau', \sigma')$ *simulates* hTAM system $\Gamma_h = (T, \tau, \sigma)$ if there's a mapping $\phi : T' \times [0, 2\pi) \times \{R, R'\} \rightarrow T$ of orientations of tiles (specified by an angle in $[0, 2\pi)$ and one of two reflections R or R') in T' to tiles in T such that for every bond graph G' generated by an assembly produced by Γ_f , there's a bond graph G produced by an assembly of Γ_h , iff G' yields G when each node is mapped via ϕ . Also, for each assembly A'_f produced by Γ_f as a single tile addition from assembly A_f , an assembly A'_h in Γ_h equivalent to A'_f via ϕ can be produced from A_h (equivalent to A_f via ϕ) by a single tile addition, and vice versa. (Namely, the exact same assemblies are producible in both systems, in exactly the same orderings of tile additions.)

Simulating aTAM systems with hTAM systems An assembly A_h over hexagon tiles T_h is a *valid c -block representation* for odd, positive integer c and partial function $\phi : T_h \rightarrow T_a$ if (1) A_h is evenly divisible into $c \times c$ blocks of tiles as shown in Figure 2a, and (2) $\phi(x)$ is defined for hex tile x in A_h iff x is at the center of a $c \times c$ block.

For a c -block representation A , define the c -bond graph of A to consist of a vertex for each center hex tile x of each $c \times c$ block with node label $\phi(x)$. Two hex tile vertices are connected by an edge if there exists a connected path of bonded hex tiles connecting the two vertices of length exactly c that consists of purely north jumps, or south jumps, or southeast jumps, or northwest jumps.

An hTAM system $\Gamma_h = (T_h, \tau_h, \sigma_h)$ simulates aTAM system $\Gamma_a = (T_a, \tau_a, \sigma_a)$ at scale c for positive, odd integer c , if there's a partial function $\phi : T_h \rightarrow T_a$, such that (1) every tile in any producible assembly of Γ_h which is of size greater than $c^2 - 1$ is within distance $d \leq c$ from a tile x for which $\phi(x)$ is defined, and (2) there's a producible assembly A_h of Γ_h that's a valid c -block representation for function $\phi(x)$ with c -block bond graph G_h iff there's a producible assembly A_a

of Γ_a with the same bond graph G_h . Further, for each producible assembly A_a of Γ_a which grows into some A'_a via a single tile addition, there are equivalent c -block representation assemblies A_h and A'_h of Γ_h such that A_h grows directly into A'_h via some number of tile additions. Note: during several tile additions, namely those which don't fill positions where $\phi(x)$ is defined, the assembly of Γ_h will still map to A_a . Vice versa, for any pair of assemblies A_h and A'_h of Γ_h such that A_h grows into A'_h via a single tile addition and the c -block representations of A_h and A'_h map to different assemblies in Γ_a , there exist assemblies A_a and A'_a of Γ_a such that A_a grows into A'_a via a single tile addition.

Simulating aTAM systems with pfbTAM systems We define a c -scaled simulation of an aTAM system by a pfbTAM system by mapping $c \times c$ blocks within pfbTAM assemblies to aTAM tiles, where this mapping reads rotations and reflections of pfbTAM tiles in the blocks. The following definition is based on the more formal definition of [6].

A pfbTAM system $\Gamma_f = (T', \tau', \sigma')$ *simulates* an aTAM system $\Gamma_a = (T, \tau, \sigma)$ at scale $c \in \mathbb{N}$, if both systems have equivalent *production* and *dynamics* under a representation function ϕ defined as follows.

(1) Production: there is a mapping $\phi : (T' \cup \{\text{empty}\} \times [0, 2\pi) \times \{R, R'\})^{c^2} \rightarrow T \cup \{\text{empty}\}$ of $c \times c$ blocks of tiles from T' and possibly empty locations (where ϕ is defined on the orientations of those tiles, specified by a rotation angle in $[0, 2\pi)$ and one of two reflections R or R') to tiles in T (or empty locations) such that for every producible assembly π in Γ_f there is a producible assembly α in Γ_a where $\alpha = \phi^*(\pi)$, and for every producible assembly α in Γ_a there exists a producible assembly π in Γ_f where $\alpha = \phi^*(\pi)$ (here ϕ^* denotes the function ϕ applied to an entire assembly, in the most obvious block-wise way). We also require that π maps *cleanly* to α under ϕ^* , that is, for all non-empty $c \times c$ blocks b in π it is the case that at least one neighbor of $\phi(b)$ in $\phi^*(\pi)$ is non-empty, or else π has at most one non-empty $c \times c$ block. In other words, π may have tiles in $c \times c$ blocks representing empty space in α , but only if that position is adjacent to a tile in α .

(2) Dynamics: if there exist producible assemblies α and α' in Γ_a such that $\alpha \rightarrow_1 \alpha'$ (growth by single tile addition), then for every producible π in Γ_f where $\alpha = \phi^*(\pi)$ it is the case that there exists π' such that $\pi \rightarrow_* \pi'$ (growth by one or more tile additions) in Γ_f where $\alpha' = \phi^*(\pi')$. Furthermore, for every pair of producible assemblies π, π' in Γ_f , if $\pi \rightarrow_* \pi'$, and $\alpha = \phi^*(\pi)$ and $\alpha' = \phi^*(\pi')$, then $\alpha \rightarrow_* \alpha'$ for assemblies α, α' in Γ_a .

2.4 Plane tiling

Plane tiling systems, such as Wang tiles [22] and Robinson tiles [17] consist of sets of shapes (called *tiles*), placed on a regular lattice, so that they cover the entire plane in an infinite arrangement. A *plane tiling system* (S, L, T, C) is a set of tiles S , specified both with shape and (optionally) color patterns on the boundary of each shape, a square or hexagonal lattice L , a set of transformations T that the tiles can undergo that necessarily includes translation (the tiles must translate to locations throughout the plane) and optionally rotation and reflection, and a set of tile adjacency constraints C that requires that color patterns on adjacent tiles must either match or be paired in complementary pairs.

A *plane tiling family* (T, C) consists of all plane tiling systems where the tiles are permitted to undergo a set of transformations and coincident tile edges must obey a set of tile adjacency constraints. The set of transformations include translation and a set $T \subseteq \{t_r, t_f\}$ of optional transformations: rotation and reflection, denoted t_r and t_f , respectively. The tile adjacent constraint

$C \in \{c_c, c_m\}$ is either that adjacent edges of tiles must match ($C = c_m$) or be paired with their complementary geometry/color pattern ($C = c_c$). We also define a *nearly-plane tiling system* to be a tiling system with the relaxed constraint that tiles need only be placed at every lattice location and touch neighbors specified by the lattice graphs, but need not fill the plane.

3 Low-Strength Hexagons Simulate High-Strength Squares

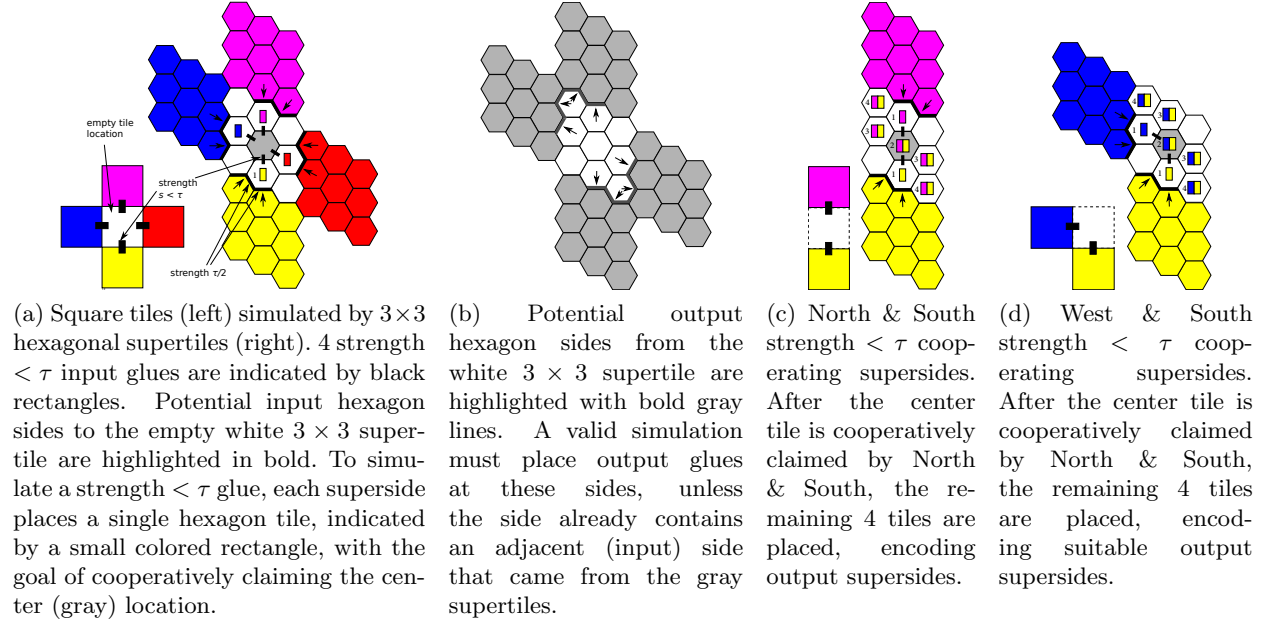


Figure 2: Simulating strength $< \tau$ squares with a temperature τ hexagon system with strength $< \tau$ glues. All hexagon glues are of strength $\lceil t/2 \rceil$, except glues bordering the center tile whose strengths are those of the simulated square input glues.

The following lemma states that any temperature τ aTAM system can be simulated by a temperature τ hTAM system that uses only glues of strength $< \tau$. In this section we present the details of the simulation and an argument for its correctness.

Lemma 3.1 *For any aTAM system $\Gamma = (T, \tau, \sigma)$ with $|\sigma| = 1$ and $\tau > 1$, there exists a hexagon assembly system $\Gamma' = (H, \tau, \sigma')$ that simulates Γ and has the property that all glues in the hexagonal-tile set H are of strength $< \tau$. Also, $|H| = O(|T|^2)$, $|\sigma'| = 3$ and the simulation has a constant scale blowup factor of 3×3 .*

Proof: (Lemma 3.1)

Representation Let $\Gamma = (T, \tau, \sigma)$ be any aTAM system. We create an hTAM system $\Gamma' = (H, \tau, \sigma')$ that simulates Γ at a scale factor of 3, i.e., each tile from T in an assembly of Γ is represented by a 3×3 square of 9 hexagonal tiles from H in an assembly of Γ' . Each such 3×3 block in Γ' is a *supertile*. The hexagonal plane is logically divided into supertile blocks (e.g., see

Figure 2 for 5 supertile hex blocks simulating 5 square tile locations) so that each supertile has 4 supertile neighbors (N, E, S, and W), which is partially accomplished by “ignoring” the northeast edge of the northeast tile and the southwest edge of the southwest tile of each supertile. For each $t \in T$, there are several supertiles in Γ' that represent t ; there must be a way to map an assembly in Γ' to an assembly in Γ . This is done with a mapping function that maps each supertile to some t by identifying its center tile. For each $t \in T$ there will potentially be several tile types $h \in H$ which can be placed as a center tile of a supertile and map the entire supertile to that t , but for each such h , there is exactly one t to which it maps. Furthermore, if a supertile is incomplete and does not have a center tile placed, then it maps to an empty location in Γ .

Simulation overview The formation of a (non-seed) supertile s begins from its outside, initiated by tile attachment from the adjacent sides of existing supertiles that serve as “inputs” to s . The information about each simulated input glue g is conveyed to locations adjacent to the center location of s by a single tile or pair of tiles specific to the direction of the input and the strength of g . If the strength of g is $< \tau$, exactly one tile is placed with one side adjacent to the center position. If the strength of g is τ , two tiles are placed, providing two such sides. Since H contains no τ -strength glues, every tile placement must be the result of cooperation between the glues of at least 2 tiles, which serve as input glues to the newly placed tile. Thus, if g ’s strength is $< \tau$, it will only supply one of the necessary sides which can allow a center tile to be placed, and that will only be able to occur if sufficiently many additional input sides place tiles adjacent to the center to provide the necessary cooperation. However, if g ’s strength is τ , it will be possible for the tiles representing g to result in the necessary 2 sides adjacent to the center position, allowing the center tile to be placed (as would be expected, since in Γ the presence of a single τ -strength glue in a neighboring position is enough to allow the placement of a tile). Note that due to the fact that the center hexagonal tile has only 6 edges, in order to allow each input path the potential of acquiring two edges adjacent to the center, the south and west pair of input sides and the north and east pair each share a position among the two sides which is “competed” for in the case that both directions represent input sides with τ -strength glues. Only one of each pair has the potential to acquire both positions and thus perhaps eventually claim placement of the center tile. This is consistent with the simulation of Γ since, if an untiled position has multiple neighbors with τ -strength glues, any of those neighbors can potentially independently direct the placement of the new tile.

Finally, after the center position is tiled, any locations adjacent to the center which weren’t used as paths for input glues are then tiled by tiles which convey the output glues consistent with the tile t being simulated to the sides which will simulate output sides.

Figure 2a shows four square tiles arranged around an empty white center square, and the south square tile (yellow) has a strength $s < \tau$ glue g exposed to its north. Input glue g , and its strength s , are encoded at the 3 southmost bold hexagon edges in Figure 2a, these are the *input sides* to the supertile, and each is of strength $\lceil \tau/2 \rceil$. The four sets of hexagonal input sides to the supertile are indicated with bold lines (and arrows) in Figure 2a. It follows that the supertile’s *output sides* are as indicated in bold in Figure 2b.

The square tile type that is simulated by a supertile is defined by the hex tile type that is placed at the center of the supertile (shown in gray in Figure 2a). Sides of supertiles, or *supersides*, compete and/or cooperate to claim this center tile location and thus set the identity of the forming supertile (in terms of the square tile type which it is simulating).

Strength $< \tau$ There are two types of bonds to simulate, strength $< \tau$ and strength τ . We first consider strength $< \tau$ bonds. Here, input supersides first attempt to place a single hex tile at one of the 4 locations indicated by the colored rectangles in Figure 2a. Then, the goal is for supersides to cooperate to place a tile at the center gray location. The tile placed at the center defines the square tile type simulated by the supertile. A superside encoding a strength s square glue will advertise a strength s glue to this center tile position as shown in Figure 2a. All other glues in the supertile are of strength $\lceil \tau/2 \rceil$.

Figures 2c and 2d show two specific examples where two superside cooperate to place a tile at the center. Numbers indicate precedence of placement within a distance 1 neighborhood, specifically, hex tile 2 can not be placed until the two neighboring tiles numbered 1 have been placed. All hex glues (in the white region) are of strength $\lceil \tau/2 \rceil$, except for two glues: the pair of cooperating tiles with label 1 expose glues of identical strength to their simulated square edge. The other cases of 2 cooperating supertiles, that encode strength $< \tau$ square sides, are symmetric with Figures 2c and 2d. The case of 3 or 4 cooperating supersides can be understood from Figure 2a. Specifically, for the case of 4 cooperating supertiles, that encode strength $< \tau$ square sides, the center gray hex tile that encodes the simulated square tile type can be placed via cooperation of all 4 supersides. When there are 3 cooperating supertiles, that each encode strength $< \tau$ square sides, we proceed similarly: 3 hex tiles cooperate to place the center tile, then tiles are cooperatively placed to tile the output hex supersides. It is straightforward, but tedious, to see that everything works in the presence of mismatches: the essential idea is that a mismatching superside does not cooperate in the placement of the center tile, and furthermore does not block the placement of any output tiles (except at its own superside).

Strength τ Simulation of strength τ bonds is illustrated in Figure 3. One of the main differences with the $< \tau$ case is that a strength τ superside must be able to claim the center hex tile without cooperating with other supersides. Hence the strategy is to tile two positions adjacent to the center position (Figure 3a), and let both of these advertise a strength $\lceil \tau/2 \rceil$ glue to the center tile location. Another trick used here is for strength τ supersides to share positions where they place their tiles: for example, in Figure 3a) it can be seen that both south and west will try to place a tile at the south-west hex tile location within the supertile. This trick is used to ensure that output paths are not blocked.

Seed structure Since supertiles are able to grow from existing supertiles in a way which simulates the tile attachments of Γ , we now need to define the seed structure. Since H can contain no τ -strength glues, it is impossible for Γ' to be singly-seeded, i.e. $|\sigma'| > 1$. In fact, for σ' to be τ -stable, $|\sigma'| \geq 3$. Therefore, to construct σ' we create the hexagonal tiles necessary to form a version of the supertile which represents the single tile of σ where all sides are output sides, and create σ' by combining 3 of those tiles in a τ -stable configuration. From this seed, the full supertile will be able to form and the full simulation will be able to proceed.

The seed assembly S of Γ' consists of 3 hexagonal tiles that are stable at temperature τ , and that grow into a supertile that encodes the seed tile s of the square system Γ . This supertile can be hard-coded using 9 unique tile types that use strength $\lceil \tau/2 \rceil$ glues. Note that the square system seed tile can appear anywhere in an assembly, and not just at the seed location. Such other versions of the seed supertile do not use these 9 unique tile types, and simply use the standard tiles described in the construction above. We need to show that a partially formed seed should not be able to

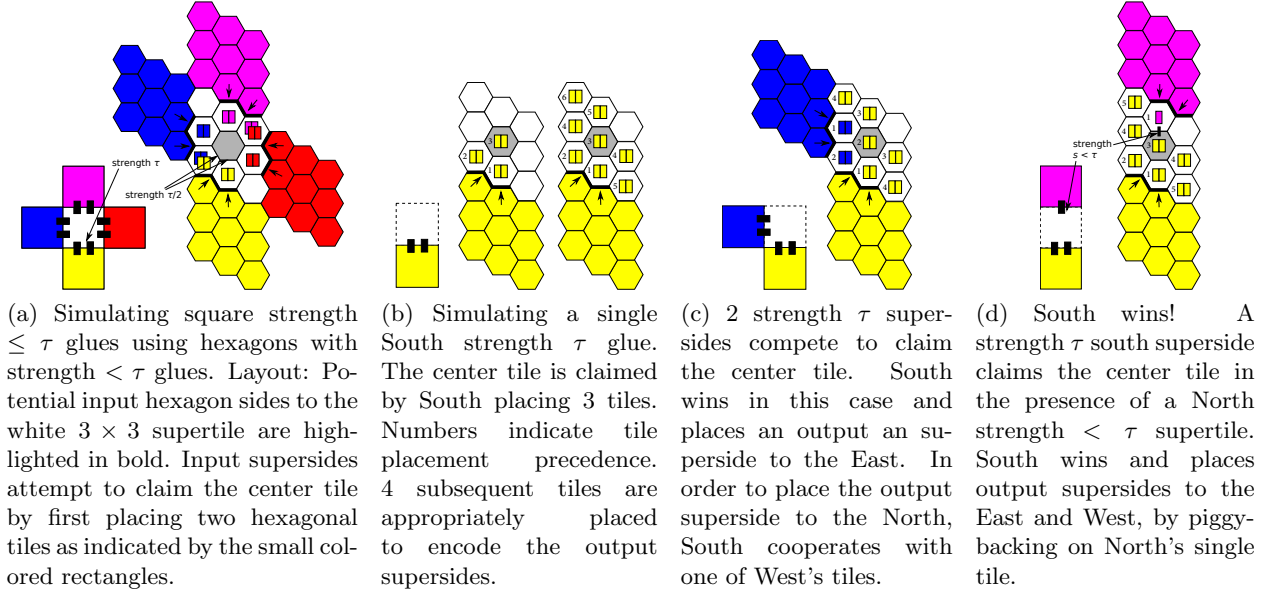


Figure 3: Simulating square strength $\leq \tau$ glues using hexagons with strength $< \tau$ glues.

initiate growth from one superside, that can grow a sequence of supertiles that come back and block the “filling out” of the seed supertile. This can be prevented by having the 3-tile hexagon seed structure consist of a triangle of hexes at the center tile, bottom left and bottom center locations, of the 3×3 supertile. It can be seen, by examining Figure 4(a), that this 3-tile seed must grow to size 5 before it can tile an output superside in such a way that initiates growth of a new supertile. At this point, 2 of the 4 output supersides are tiled, and so they can not be blocked. Of course, these output sides could produce supertile tentacles that could grow around and attempt to block the other 2 output supersides by tiling inside the 3 “outputting locations” of the seed supertile location. For this to happen such a tentacle needs to grow a supertile adjacent to one of the 2 remaining (untiled) supersides, hence that side is blocked in a way that is completely valid. However the tentacle can place 1 or 2 hexagon tiles inside the seed supertile at this point (at input locations), but then its growth is terminated because it can not claim the (already tiled) center tile location. Furthermore, there is one untiled corner site remaining in the supertile that is adjacent to tentacle tiles. However, this is at an output-only location in the superseed location, and so the tentacle presents 0-strength glues there and so can not tile that position. There remains a tiling path (using both generic, and seed supertile-only glues) to tile the remaining output superside.

3.1 Correctness of Hexagon Construction

In order to show that a supertile correctly simulates a square tile, we show that if a square binding event should be simulated, the relevant combination of input supersides leads to a valid placement of output supersides, and otherwise no output supersides should be presented. It can then be seen from Figures 2a and 2b, that appropriately tiled output supersides implies that those supersides are ready to act as input supersides to the adjacent supertiles. We consider 4 main cases, where there are 1, 2, 3 or 4 input supersides present (of course, not all input supersides may end up

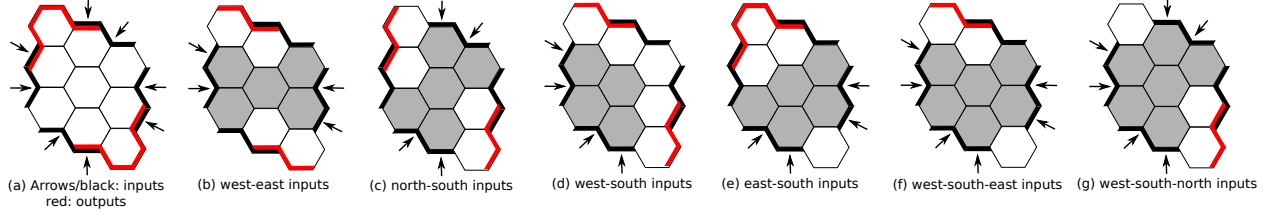


Figure 4: Hex inputs (bold black, with arrows), and hex outputs (red), for the four supersides. The shaded gray area is a superset of the tiled area upon placement of the center hex tile, irrespective of the strength of square glue being simulated. the figure covers a number of 2- and 3-sided binding cases. In each case we see there is a tiling path to the outputs that can be tiled using strength $\lceil t/2 \rceil$ hex glues.

contributing to the supertile choice, e.g. 1 or more may mismatch).

When there is exactly 1 strength $< \tau$ superside present the center tile is not claimed, and so no square tile type is simulated (Figure 2a). When there is exactly one strength τ superside present the center hex tile is claimed, as shown in Figure 3b. There is a path to appropriately place the 3 output supersides (using strength $\lceil \tau/2 \rceil$ glues).

Next we consider the case of 2 input supersides. It can be seen from Figures 2c, 2d, 3c and 3d (and their rotations) that there are scenarios where either one of the 2 sides can tile a path to the center tile, and the winning superside is determined by the order of tile placement. In other words, each superside has an opportunity to claim the center tile, and no superside is blocked until the center tile is placed. Figures 4b–e, and all rotations of these, can be used to argue that there is a path to tile the output supersides appropriately. Immediately upon claiming the center tile, the shaded area shows the total area tiled when there are 2 input supersides of strength τ , and it shows a superset of the tiled area when there are input supersides of strength $< \tau$. In each case there is a path to tile the remaining 2 output supersides: specifically there is a path to tile the 2 red output locations using strength $\lceil \tau/2 \rceil$ glues.

For the case of 3 input supersides, Figures 2a and 3a can be used to reason that the center tile can be placed, and furthermore that through cooperation between supersides, all supersides have the opportunity to individually claim (strength τ supersides) or cooperatively claim (strength $< \tau$ supersides) this tile location. Figures 4f and 4g (and 2 rotations of these) illustrate the situation upon claiming of the center location (the shaded area represents a superset of the tile locations at this point). Again, in each case there is a path to tile the remaining output superside, specifically there is a path to tile the red output location using strength $\lceil \tau/2 \rceil$ glues.

For the case of 4 input supersides, Figures 2a and 3a, illustrate the various ways in which the center tile could be claimed.

Of course, it could be the case that the input supersides encode mismatching square sides that correspond to no tile placement, and this is handled by the construction as follows. (a) A superside encodes a glue that has no corresponding opposite side glue: here no hex tiles are at all placed by the corresponding “input superside” (the relevant supersides in Figures 2a and 3a would have *no* tiles placed with colored rectangles). (b) superside encoded insufficient strength (or mismatching glues) for an attachment: in this case the tiles denoted using colored rectangles in Figure 2a advertise insufficient strength, or mismatching glues, to the center tile position.

In order to generate all of the hexagonal tiles that are needed for the different versions of each supertile, it is sufficient to create $O(|T|^2)$ hexagonal tiles. Intuitively, this allows for the creation of

a unique hexagonal tile for each combination of the aTAM tile types represented by its two input tiles. \square

Furthermore, it is straightforward to see from the above construction that an aTAM system with seed σ where $|\sigma| \geq 1$ (i.e. a seed assembly consisting of many tiles) is simulated by an hTAM system where the $9|\sigma|$ hexagonal tiles simulating the aTAM seed assembly are appropriately placed to represent that seed assembly. Thus the following Corollary holds.

Corollary 3.2 *For any aTAM system $\Gamma = (T, \tau, \sigma)$ with $|\sigma| \geq 1$ and $\tau > 1$, there exists a hexagon assembly system $\Gamma' = (H, \tau, \sigma')$ that simulates Γ and has the property that all glues in the hexagonal-tile set H are of strength $< \tau$. Also, $|H| = O(|T|^2)$, $|\sigma'| = 9|\sigma|$ and the simulation has a constant scale blowup factor of 3×3 .*

4 Encoding Glues in Geometry

We utilize the idea (similar to [8]) of using small surface geometries on polygonal tiles to encode additional information. These geometries have two canonical types: indentations and protrusions, which we call *dents* and *bumps* for convenience. Such geometries are also used extensively in subsequent sections to both encode information and enforce constraints on how tiles can bond. However, here we merely describe how to perform a simple modification on a polygonal tile system to reduce the number of glues in the system to one by encoding these glues in bump-and-dent geometry.

Modification Given a polygonal tile set T with glue set $G = \{g_1, g_2, \dots, g_m\}$, we define a *glue geometry* consisting of a sequence of m dents and m bumps. Each side of every tile in T is modified to include this geometry oriented around the midpoint of the side (dents on the counterclockwise (CCW) side, bumps clockwise (CW)), with the original glue of the side removed and a set of small *subsides* with a common strength-1 glue created within the glue geometry. (From now on, we are working with a slightly generalized pfbTAM model that permits this kind of glue placement.) For a side with strength- k glue g_i , these subsides are created by dividing the tip of the i th leftmost dent and i th rightmost bump into $\lfloor k \rfloor$ short sides, each with a strength-1 glue (see Figure 5). The subsides have non-repeating lengths unique to g_i . If k is odd, a short side between the bumps and dents of a length unique to g_i with the common strength-1 glue is also created.

Correctness First, because the glue geometries are small geometries placed at the midpoint of each side and matching only each other, sides can only attempt to bond with coincident midpoints, as they did in the unmodified tile set, with the k dents of each glue geometry accepting the k bumps of the other. Also note that side pairs are not prevented from coming together in this way, even if the sides do not bond, as all glue geometries have matching geometries with each other.

Next, any pair of unmodified sides that bond with positive strength have identical glues. In the modified tile set, the glue geometries of these sets both have a sequence of subsides with strength-1 glues on the tips of the i th leftmost dent and i th rightmost bump (and possibly on a short side spanning the midpoint of the unmodified edge). As a result, a total of $2 \cdot \lfloor k/2 \rfloor$ strength-1 bonds are formed, plus 1 if k is odd. So bonds with total strength k are formed in the modified tile set.

Finally, any pair of sides in the original tile set with distinct glues do not form a positive strength bond, but may become coincident and form a “strength-0 bond”. In this case, the glue geometries

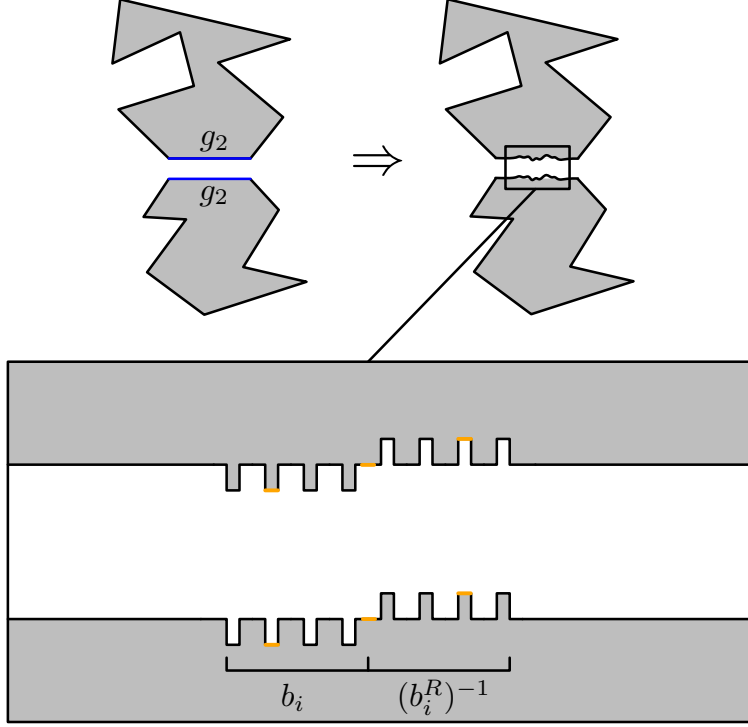


Figure 5: Converting a side with a strength-5 glue g_2 (blue) to a set of smaller sides using a common glue (orange). The original glue set has 4 glues in this case.

have matching geometries as mentioned before, but the glues in the glue geometries for both sides lie on distinct bumps and dents, and do not meet.

All that remains is to show that no unwanted positive-strength bonds can occur in the modified tile set. The cases in which the two corresponding sides in the original tile set are coincident has already been considered. Recall that the lengths of the subsides are unique to each g_i . As a result, the only sets of subsides that can form positive-strength bonds are those corresponding to the same g_i . Because the lengths are non-repeating, the only matching sequence of subsides occurs when two bumps both have subsides with glues corresponding to the same glues *and* these glues appear mirrored, as when one tile has been reflected or “flipped”.

Since the glue geometry is small and the constructions using glue geometry either use large bump and dent geometry to forbid tile reflection (as in Section 5) or exist in models that forbid reflection, such an orientation of subsides cannot occur. So all bonds occurring in the modified tile set correspond to valid bonds in the original tile set with the same total strength.

5 Self-Assembly With a Single, Rotatable Tile Type

Now we introduce the use of *rotation* of a single polygonal tile to encode a set of multiple simple tiles. We show that hTAM systems that do not contain τ -strength glues can be simulated by a pfbTAM system consisting of a single (nearly) regular convex polygon (with small surface geometry) that may rotate and flip. The hTAM systems are simulated at scale 1 in a natural way: the rotation

angle of a tile indicates the hexagonal tile it simulates. The use of small surface geometry (as used in Section 4) constrains the possible orientations in which the tile can attach to the seeded assembly to match only those that correspond to valid hexagon tiles.

First we define the class of polygonal tiles utilized for this construction. Then we present the full details of the construction for simulating any hTAM system without any τ -strength glues using a single-tile pfbTAM system. Because there are no τ -strength glues in the simulated hTAM system, any such system assembling a non-trivial assembly requires a multi-tile seed. The construction we present uses an equivalent seed in the simulating pfbTAM. We then use this construction to prove Theorem 5.2.

Finally, we describe a modification to the construction that eliminates the need for a multi-tile seed in the simulating pfbTAM (by using a τ -strength glue), thereby making the pfbTAM system self-seeding. In this case no seed assembly must be created as input for the system, but instead one individual copy of the single tile type will assume the role of seed for each assembly which forms by then growing from it), and thus prove Theorem 5.3.

5.1 n -gon tiles

A *regular n -gon* is an equilateral and equiangular convex polygon with n sides, for $n \in \mathbb{N}$ and $n \geq 3$ with each side length 1. For this construction, we define a family of polygonal tiles we call *n -gon tiles* as a set of unit regular n -gons with the addition of small-scale surface protrusions (*bumps*) and indentations (*dents*), as used to encode glues in Section 4.

Side geometry The geometry of each side of a tile consists of the placement of a bump and dent pair in one of two locations on the polygon side, either towards the clockwise or counterclockwise end of the side. See Figure 6 for an example of the geometries on a polygonal tile and details about the sizes and placement of bumps and dents. The interior angles between the edges of the n -gon are $\alpha = (1 - 2/n) \cdot 180^\circ$. Each bump consists of a rectangular protrusion, extending perpendicularly from the tile's side, while each dent is a rectangular concavity in the side. Bumps and dents have identical size and are placed adjacent to each other. Let h and w be the height and width of the bumps and dents, and d be the distance separating them, with dents preceding bumps when travelling counterclockwise along the boundary of the tile. Set $h = d \cdot \tan(180^\circ - \alpha)/2$, $w = d/4$, $d = 1/10$, where *CCW* and *CW* geometries are placed distance d from the counterclockwise-most and clockwise-most ends of the polygon edge, respectively.

We claim that these geometries have the following properties: 1. dents in the interior of the tile do not collide with each other (see the dents in the detail of Figure 6 for an example of dents that are close to a collision), and 2. for any two tiles with a pair of coinciding sides, the bump on any non-coinciding side of a polygon does not intersect the other polygon.

Property 1 holds as long as the dents in consecutive sides with *CCW* and *CW* geometries do not collide, because all other pairs of dents are further apart. Because the fewest sides that an n -gon could have is 6 (which would occur if $c = 1$), the smallest that α can be is $(1 - 2/6) \cdot 180^\circ = 120^\circ$. Then because $h = d \cdot \tan(180^\circ - \alpha)/2 \leq d \cdot 1.74/2 < d$, it is impossible for two dents to intersect.

Property 2 holds provided that a *CW* bump on side s_i is contained entirely within the half-plane formed by the supporting line (the dotted line in Figure 6) of side s_{i-1} and containing the rest of the tile. Because $\alpha \geq 120^\circ$ for all n -gon tiles with $n \geq 6$, then $\tan(180^\circ - \alpha) \leq 1.74$. The bump is distance d away from the vertex and has height h , so since $h/d = \tan(180^\circ - \alpha)$, the bump does not intersect the supporting line.

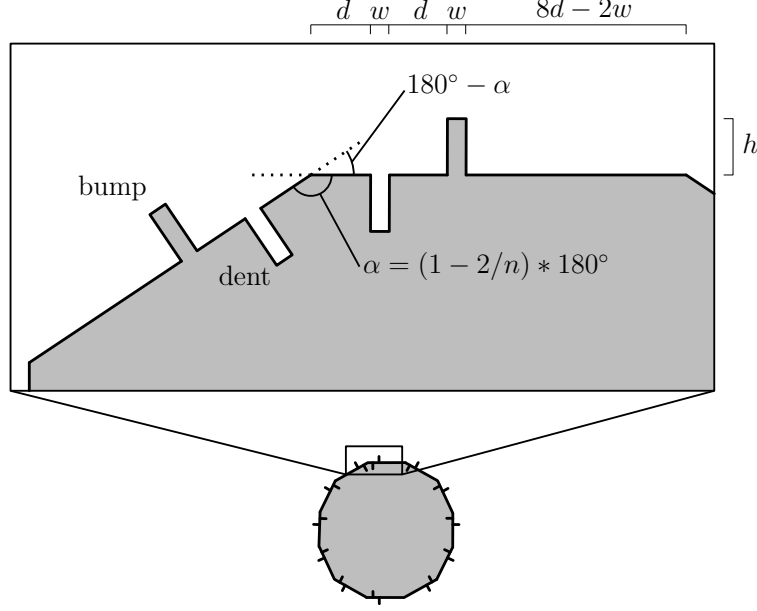


Figure 6: A pair of adjacent sides of an n -gon tile.

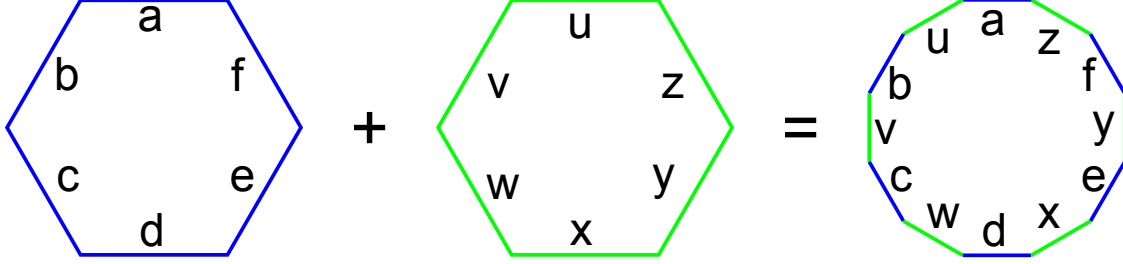
Each n -gon tile $t \in F$ has a sequence of sides oriented CCW and starting with s_0 , and an identity rotation and flip with s_0 horizontal at the top of the polygon. Specifying each side of t requires a glue label and location for the bump-dent geometry (either at the CW or CCW end of the side).

5.2 Simulating the hTAM

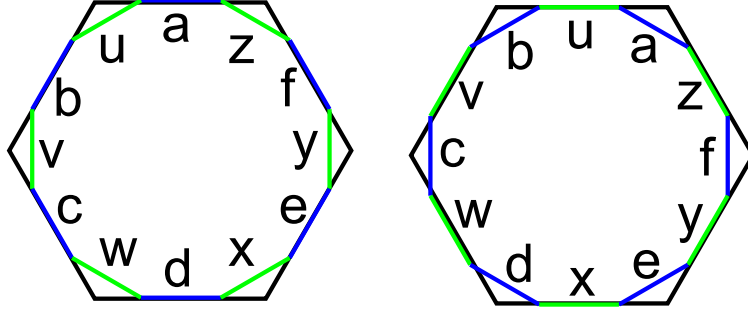
Lemma 5.1 *For any hTAM system $\Gamma_h = (T, \tau, s)$ such that T does not contain any τ -strength glues and $|s| = 3$, there is a free-body self-assembly system $\Gamma_f = (F, \tau, s_f)$ with $|F| = 1$ and $|s_f| = 3$ that simulates Γ_h .*

Proof: (Lemma 5.1) To prove Lemma 5.1, we provide the following construction to create Γ_f from the definition of Γ_h . Let $c = |T|$, the number of tile types in the simulated system. F will consist of a single n -gon tile type where $n = 6c$, created by utilizing the following transformations:

1. Create a new (hexagon) tile set T' as follows: for each $t \in T$, add a new tile type t' in T' where every glue is also labeled according to the pair of opposite sides (1 for N/S , 2 for NW/SE , 3 for SW/NE) it is found on. For instance, a set of six counterclockwise glues a, b, a, c, d, b on t would become $a_1, b_2, a_3, c_1, d_2, b_3$. This ensures that a glue appearing on a side s of a tile is only able to bind to another glue which is on either the same or opposite side of a tile.
2. Let $p \in F$ be an n -gon tile type. Orient p in its identity orientation, with s_0 placed horizontally as the uppermost side. For sides $s_j, 0 \leq j < c$, assign the North glue of the j th tile in T . Similarly, for sides $s_{j+c}, s_{j+2c}, \dots, s_{j+5c}$, assign the NW, SW, S, SE, and NE glues of the j th tile in T . Thus, each consecutive group of $n/6$ sides, starting from s_0 , represents the sides of one particular direction of the $c = n/6$ tiles in T . This results in the sides of p containing



(a) Assuming T consists of 2 tile types, the blue and green, the 12-gon tile on the right would be created.



(b) The 12-gon inscribed in a hexagon, with the orientation which represents the blue tile (left), and the orientation representing the green tile (right).

Figure 7: Example creation of a 12-gon tile type.

the 6 glues of each tile type in T in positions such that they coincide with the 6 sides of the smallest hexagon into which p can be inscribed (in some orientation). See Figure 7 for an example.

3. The final modification to p adds geometries consisting of an adjacent bump and dent pair per side at one of two locations (CCW or CW) along the edge. Assign the geometry locations for $s_j, j < 3c$ to be CCW , and the geometry locations of for $s_j, 3c \leq j$ to be CW . Thus the geometry on the consecutive sides representing the North, Northwest, and Southwest sides of tiles in T has one configuration, and the geometry on the other sides is distinct.

To form the seed s_f , we define s_1, s_2 , and s_3 , where $s_1, s_2, s_3 \in T$, as the three tile types that form s . We then take 3 copies of the n -gon $p \in F$, denoted by s_{p1}, s_{p2} , and s_{p3} . Orient s_{p1} so that if it is inscribed within the smallest hexagon that contains it, and that hexagon is oriented such that it has north and south sides, then the 6 sides of s_{p1} that coincide with the bounding hexagon represent the 6 sides of s_1 in that same orientation. Orient s_{p2} and s_{p3} similarly with respect to s_2 and s_3 . Now place s_{p1}, s_{p2} , and s_{p3} adjacent to each other (without additional rotation) in the same relative configuration as s_1, s_2 , and s_3 occur within s . Since the three adjacent pairs of edges in s bind to form a τ -stable assembly, by the construction of p , the edges of the 3 copies of p in their current rotations and positions are guaranteed to also bind to form the analogous τ -stable seed s_f .

Assembly proceeds by correctly oriented copies of p binding with at least 2 edges of n -gons in the growing assembly (since there are no τ -strength glues). As it has been shown that correct tile bindings will occur, we now show that incorrect bindings cannot occur. Incorrect bindings must be

prevented in the cases of 1. rotations that don't correspond to hexagonal tiles which should bind being allowed to bind, and 2. copies of p which have flipped over (and possibly also rotated) being allowed to bind.

To correctly simulate non-rotatable hexagons, each side of p must only bind to other sides that simulate the opposite side of a hexagon, i.e., sides of p that represent N, NW, SW, S, SE, and NE sides of tiles from T must bind only to sides of p that represent S, SE, NE, N, NW, and SW sides, respectively. As previously mentioned, this creates the pairs of opposite directions (N/S , NW/SE , and SW/NE). Binding to a side representing a direction not in the correct pair is ensured by Step 1 in the creation of p . Correct binding to a complementary side is allowed by the existence of the correct glues and also the definition of the geometries.

Incorrect binding is prevented by a combination of geometries and direction-specific glues. For each orientation corresponding to a hTAM tile, there are 11 others that are “incorrect”, but related to the tile: they are the 11 orientations that correspond to rotations or flips of the hexagonal tile. Because the simulating pfbTAM system uses distinct glues for each pair of opposite directions, and at least two bonds are needed for a tile to attach, there are at most four possible orientations of the 12 total that can possibly attach: 1. the valid orientation, 2. a 180° rotation from valid, 3./4. a flip about the direction pair containing the two bonds, and a 180° rotation of this flip. Note that 3. and 4. only exist if only two bonds are needed to attach and they lie in the same direction pair, otherwise only 1. and 2. are possible. Figures 8a, 8b, and 8c show that Cases 2, 3, and 4 are all forbidden due to mismatch in geometries. \square

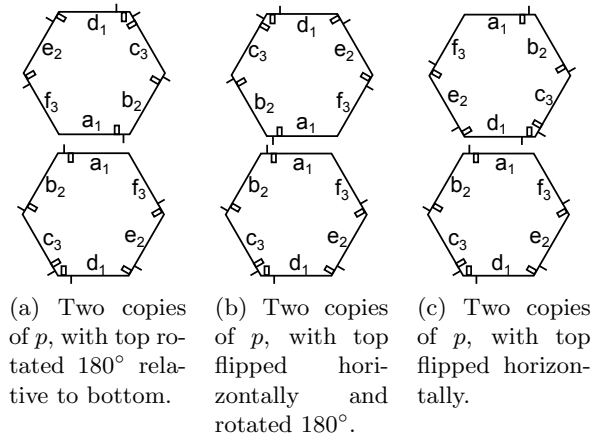


Figure 8: Cases with glues in correct opposite-direction pairs, but binding is prohibited by geometry mismatches.

Theorem 5.2 (*Universal Single Tile Simulation*) *There exists a single polygonal tile p such that, for any aTAM system $\Gamma = (T, \tau, \sigma)$ where $|\sigma| = 1$ and $\tau > 1$, there is a τ -stable seed assembly σ' such that the pfbTAM system $\Gamma' = (\{p\}, 2, \sigma')$ simulates Γ .*

Proof: By Theorem 3.1 of [6], there is a single aTAM tile set which, when properly seeded and operating at temperature $\tau = 2$, can simulate any aTAM tile assembly system. Let U be that tile

set. Let \mathcal{T} be an arbitrary aTAM tile assembly system. By Theorem 3.1 of [6] there is an aTAM system $\mathcal{T}' = (U, 2, \sigma)$ which simulates \mathcal{T} . Let $\mathcal{H} = (H, 2, \sigma_H)$ be the hTAM system which simulates \mathcal{T}' without using strength 2 glues (as constructed using the construction in Section 3 and defining the seed σ_H as the fully formed supertiles which represent σ as mentioned for Corollary 3.2). Let p be the polygonal tile type generated from H using the steps in the proof of Lemma 5.1, and σ_p be a seed assembly of properly rotated and positioned copies of p to represent σ_H . Then, it follows that $\mathcal{P} = (\{p\}, 2, \sigma_p)$ simulates \mathcal{T} . \square

As U is a fixed tile set, p is also a fixed polygonal tile with a constant number of sides. With T tiles in the simulated aTAM system, the scale factor of our simulation is $O(|T|^4 \log |T|)$, as that is the scaling factor for the simulation of \mathcal{T} by \mathcal{T}' (see [6]), and the only further scaling is a constant (i.e. 3×3) for the simulation of \mathcal{T}' by \mathcal{H} .

5.3 A Self-Seeding pfbTAM

In the construction described in 5.2, the three-hexagon-tile seed assembly in an hTAM is simulated by a three-polygon-tile seed assembly in the pfbTAM. In this section we describe an extension of this construction that also simulates any hTAM system with no strength- τ glues, but does not use a multi-polygon-tile seed. Instead it has the more traditional single-tile seed, with the tradeoff that a single strength- τ bond is used. Notice that if no τ -strength bonds exist in a pfbTAM system (or any seeded aTAM-like system), assembling any multi-tile assembly requires a multi-tile seed, as no two-tile τ -stable assemblies exist.

The construction will be similar to the simulation construction in that the tile is a polygon tile with glues on each side and small geometries. However, several additional steps are used to create a polygon tile that simulates the hexagon system and has the following property: every new polygon tile that attaches to the seed assembly during the assembly process bonds using a side that lies in a particular 60° wedge of the polygon tile. To achieve this, the techniques of [3] for their simulation results of Section 3, specifically “minimal glue sets” and “inward-outward glues”, are utilized.

Theorem 5.3 (*Self-Seeding Single Tile Simulation*) *For any aTAM system $\Gamma = (T, \tau, \sigma)$ with $|\sigma| = 1$ and $\tau > 1$, there is a pfbTAM system $\Gamma' = (F, \tau, \sigma')$ with $|F| = 1$ and $|\sigma'| = 1$ that simulates Γ .*

The io-hTAM Let $\Gamma_h = (T, \tau, s), |s| = 3$ be the hTAM system being simulated. For each hexagon tile $t \in T$, define a *minimal glue set* to be a subset of the sides of t whose glues have total strength at least τ such that any strict subset of these sides has total glue strength less than τ . Create a second hTAM system $\Gamma'_h = (T', \tau, s'), |s'| = 3$ where for each tile in T , a set of tiles in T' are created, one for each minimal glue set. For each tile created, the glue on each side in the minimal glue set is marked *IN*, and the glues of all other sides are marked *OUT*. We define the io-hTAM model to be identical to the hTAM model except that assembly proceeds under the following additional constraints: no pair of *IN* sides from distinct tiles may be adjacent, and any pair of adjacent *OUT* sides form a strength-0 bond (but are permitted to touch). Add three additional tiles to T' generated from the three tiles in s by marking one side per tile *IN* and all others *OUT* such that the these three tiles when placed in the same configuration as s form a τ -stable three-tile assembly. Define this configuration of these three tiles as s' . Define two assemblies α, α' producible by Γ_h, Γ'_h to be *equivalent* if replacing each tile α' with the tile generating it in Γ_h yields α .

Lemma 5.4 *An assembly is producible by Γ_h in the hTAM model if and only if there is an equivalent assembly producible by Γ'_h in the io-hTAM model. Moreover, every exposed side on every assembly produced by Γ'_h in the io-hTAM model is marked *OUT*.*

Proof: The proof is by induction. Recall that the seed s' is constructed such that all exposed sides are marked *OUT*. Now consider producing an assembly β' in Γ'_h by adding a tile t' to an existing assembly α' where 1. β' has an equivalent assembly β producible by Γ_h by adding a tile t , and 2. all exposed sides are marked *OUT*.

Such a tile t' exists, as the set of bonds used by t to attach to α contains a minimal glue set also sufficient to bind and T' contains a tile where this set of sides is marked *IN* (and all others are marked *OUT*). Moreover, any tile $t' \in T'$ was generated by some tile in T and if t' can bind in the io-hTAM, this tile must be able to bind in the hTAM, as its binding requirements are strictly weaker. Additionally, any tile t' attaching uses all *IN* sides to bond, so any exposed side on t' after attaching to α' to form β' must be *OUT*. So all sides of β' are marked *OUT*. \square

Implementing the io-hTAM in the pfbTAM Next, we consider implementing an io-hTAM system as a pfbTAM system with a single polygon tile. To do so, we follow the approach in the previous simulation construction with the addition of new geometry to each side to enforce the *IN-OUT* side constraints. Recall that we want two sides with *IN/OUT* markings to attach using the following rules:

- Two *IN* sides are not permitted to meet (forbidden).
- A pair with opposite marks can meet and bond according to glue strength (matching).
- Two *OUT* sides can meet but not form a positive strength bond (permitted).

Implementing this as geometry is simple: an *IN*-marked side has a small bump, while an *OUT*-marked side has a matching dent. See Figure 9 for a detailed depiction of the geometry.

In this case two sides with *IN* cannot meet due to geometry, *IN/OUT* pairs of sides can meet and can bond, and two sides with *OUT* can meet, but do not have matching geometry and thus do not form a bond. We place the bump or dent at the bisector of each side, with the height (or depth) and width equal to that of the bump and dent system implemented in the previous construction. Now construct a single-tile pfbTAM system as done previously, but with the modified hTAM and additional geometry on each side: for each hexagon tile with marked *IN* and *OUT* sides, add six sides to the polygon at 60° intervals. Each side corresponds to a side of the hexagon tile, with appropriate glue, bump-and-dent pair according to its location on the hexagon, and bump or dent according to whether it is an *IN* or *OUT* side.

Enforcing a polygon tile invariant The previous steps yield a single-tile pfbTAM system that simulates an arbitrary io-hTAM system with no strength- τ glues. Next, the polygon tile is modified so that during assembly *every* attaching tile must use a side that lives within a particular 60° arc of the tile's boundary. Recall that each hexagon tile in the io-hTAM system must use all *IN* sides (forming a minimal glue set) to attach to an existing assembly, and each hexagon tile in the io-hTAM corresponds to a set of six sides of the polygon tile spaced at 60° intervals. For each such

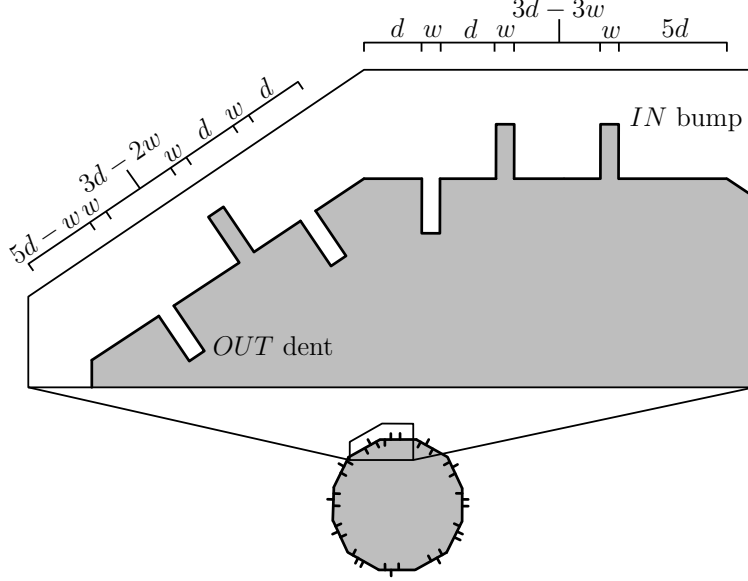


Figure 9: A pair of adjacent sides of an n -gon tile simulating a tile system in the io-hTAM model.

set of six sides, ‘rotate’¹ the set until the side lying in the interval $[0, 60^\circ)$ is one which is marked *IN*. Now the polygonal tile still may attach to simulate the hexagon tiles as before, but with the invariant that every attachment includes a bond formed by a side in the $[0, 60^\circ)$ interval arc on the boundary of the polygon.

Adding self-seeding All that remains is a modification to the simulating polygonal tile that removes the requirement of starting with a three-tile assembly. Note that during the assembly process, every attaching tile uses two sides to bond, and any side within 60° of either of these sides lies in a narrow region between the two tiles sharing the bond. So geometry prevents polygonal tiles from bonding to such a side. Additionally, because each attaching polygonal tile uses a side in its angular wedge $[0^\circ, 60^\circ)$, no polygonal tile uses more than one side from this set. We now use this geometric fact to create a self-seeding tile.

Select two hexagons in the seed s' of the io-hTAM system to use as a self-seeding two-tile assembly. Add two more sets of six sides for these two chosen hexagons, and replace the glue (which has strength less than τ) shared by these two hexagons in the three-tile seed s' with a pair of τ strength glues. For each six-side set, the side simulating the strength- τ bond is placed adjacent to the position at angle 30° (i.e. the bisector of the $[0, 60^\circ)$ wedge containing only *IN* sides) and has no *IN/OUT* geometry. All other sides are marked *OUT*.

Consider the behavior of the single-tile pfbTAM system: starting with a single seed tile, a second tile must come and attach via the strength- τ bond. From then on, any tile attaching uses at least two sides to attach, all of which must be *IN* sides, and one of which lies in a small region of the boundary containing the strength- τ bond. Once the tile is attached, the strength- τ bond becomes unusable, as it is geometrically blocked by the neighboring tile which the *IN* side in this region attached to. By induction, this remains true for assemblies produced via an arbitrary number of

¹Here rotation is equivalent to cyclic permutation.

steps.

6 Translation-Only Systems

Similar to the previous section, we discuss the power and limitations of systems composed of tiles of a single type, in terms of the aTAM tile assembly systems which they can simulate. However, while previous assemblies were composed of translated copies of the single tile type which were able to represent each of multiple aTAM tile types by being positioned in specific relative rotations to each other, here we no longer allow rotations. Instead, while assemblies are still composed of translated copies of a single tile type, the assembly can be thought of as lying within a logical rectangular grid, such that each tile has a “central body” that is contained within the bounding rectangle for its coordinate position, and the identity of the tile type from the simulated aTAM system that it is representing is determined by the position of the tile’s central body relative to that bounding box. Thus, the simulation of multiple tile types occurs by one single tile type that can assume a specific translation relative to a fixed coordinate system for each tile type that it can simulate: in summary, an aTAM tile type is encoded by the relative position of our single tile type. Therefore, in this section the term *translation* is used to describe the local translation of tiles within the logical bounding boxes of their coordinate locations, rather than the global translation represented by the positioning at the various coordinate locations.

We show that such systems can simulate computationally universal systems. This requires arbitrarily large seeds (whose size is a simple linear function of the number of timesteps in the simulated computation). We then prove strict limitations on such systems with small seeds.

6.1 Universality in Translation-Only Systems

In this section we show that for any given 1D cellular automaton, there exists a single tile, such that with an appropriate seed consisting of $O(n)$ copies of this tile, the corresponding translation-only tile system will simulate the first n steps of the given cellular automaton. The universality of 1D cellular automata thus yields a single translation-only tile that is computationally universal (using infinite seeds).

6.1.1 Definitions

In this section, for simplicity, we will refer to unit-square aTAM tiles as sitting at a 45 degree rotation from the standard orientation. In particular, we will refer to the four sides of such a square aTAM tile as the northwest, northeast, southeast, and southwest faces.

Slider Definition Consider some standard square tile aTAM system $\gamma = (T, \tau, s)$. Further, consider some total one-to-one function $X : T \rightarrow \mathbb{N}$, such that the codomain of X is a set of numbers that satisfy Lemma 6.1. Let this codomain be denoted by the variables $x_1, \dots, x_{|T|}$, such that $x_i < x_{i+1}$ for each i . Let X^{-1} denote the inverse of X . Note that by Lemma 6.1, we have that $1 \leq x_{|T|} \leq 3|T|^5$. Further, without loss of generality, assume that the leftmost, southmost tile type in s is such that X maps it to x_1 . For such a γ and X , we construct the corresponding *slider* tile, $SLDR_\gamma^X$, in Figure 10. The slider consists of 6 grey pads of unit width and length equal to $x_{|T|}$. Further, each pad has a glue assigned to positions $x_1, \dots, x_{|T|}$ from south to north, on either the west or east side of the grey pad as depicted in Figure 10. Each of the 6 grey pads are connected

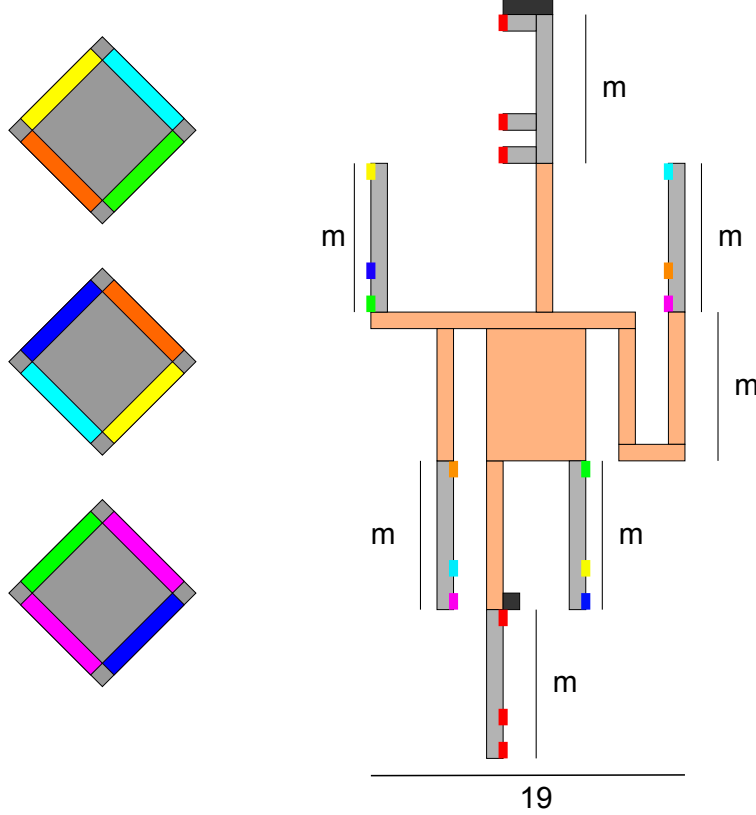


Figure 10: A *slider tile* derived from a set of square tiles T , representing each tile in T by sliding up and down into different relative translations compared to its neighbors. The slider has 6 bonding pads, each with a sequence of glues positioned according to a sequence of positive integers $x_1, \dots, x_{|T|}$ that satisfy the algebraic constraints of Lemma 6.1. The glue at position x_i on the northwest pad is the glue on the northwest face of the i th tile type from T (starting with x_1 on the southmost tip of the pad). The glues for the northeast, southwest, and southeast pads are similarly assigned. The top and bottom pads also have glues assigned at positions $x_1, \dots, x_{|T|}$, but all glue types on these pads are of the same, neutral type. In this figure, m denotes the value $x_{|T|}$, the largest integer in the sequence $x_1, \dots, x_{|T|}$. From Lemma 6.1, we know that an appropriate sequence exists such that $m \leq 3|T|^5$.

by the tan portion of the tile and the exact dimensions are provided in Figure 10. In short, the width of the tile is a constant, with the height being linear in $x_{|T|}$. Additionally, the glue type on the northwest pad at position x_i is the glue that occurs on the northwest face of the tile $t \in T$ such that $X(t) = x_i$. The glue type for the northeast, southwest, and southeast pads are defined similarly. For the north and south pads, the glue types at each position are all of the same glue type a , where a is a strength-1 glue that does not occur within the tile set T .

Slider Assembly Mapping We now discuss how an assembly of slider tile types represents an assembly over a set of square aTAM tiles T . For a pictorial description of the mapping, see Figure 11. The key idea is to place an imaginary grid of gray boxes over a given slider assembly

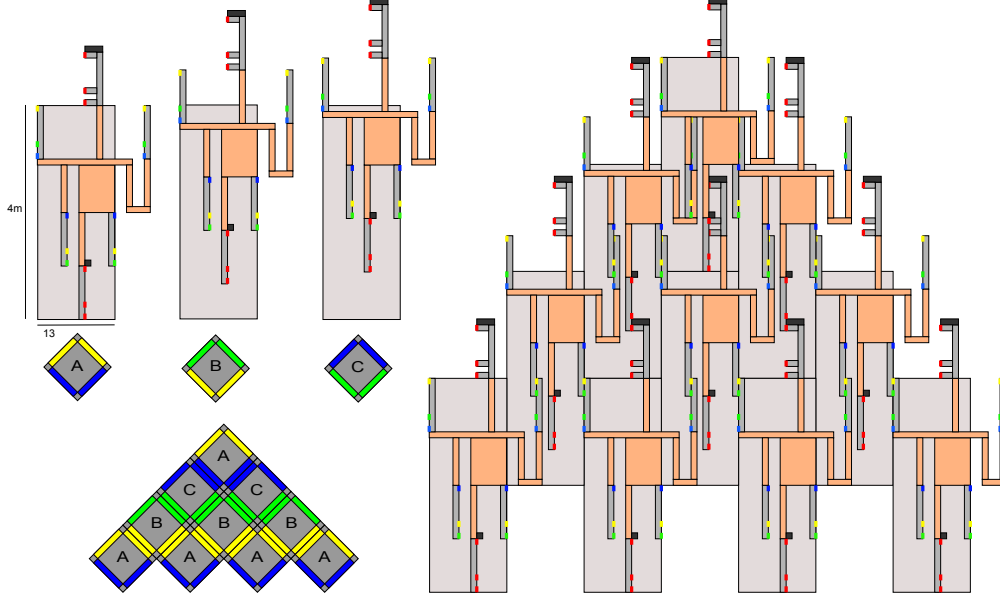


Figure 11: An assembly of sliders for a square tile set T is mapped to a corresponding square tile assembly over T by placing an imaginary grid shown by the grey background blocks in this figure. The tile of T represented by the slider in each grey block is determined by the north/south translation of the slider in the given block. In particular, the slider represents the i th tile of T if the i th glue on the northwest pad of the slider lines up with the northernmost position of the grid rectangle. (Note that the assembly shown in this example does not adhere to the double-checker-boarded constraint and so such a produced assembly would not be guaranteed to form without error in the slider construction.)

to define the position each slider is conceptually tiling, as well as the type of tile represented by referencing the relative north/south translation of the slider within the grey box. We now formally define the mapping of a slider assembly to a square aTAM assembly.

Consider a τ -stable assembly A consisting of translations of a slider tile $SLDR_\gamma^X$. Now consider the westmost, southmost slider tile in A . Assume this slider tile sits at coordinate position $(0, -x_1)$. We now define a partial mapping $f : Z \times Z \rightarrow Z \times Z \times T$, which maps slider coordinate locations within an assembly to both a 2D coordinate position and a tile type in T .

To define $f(x, y)$, first let $w = 13$ and $\ell = 4x_{|T|}$ (w and ℓ denote the width and height of the grey grid boxes from Figure 11). Define $f(x, y)$ as follows: If for integers $a, b \geq 0$ and $t \in \{x_1, \dots, x_{|T|}\}$ it is the case that $x = 2wa$ and $y = b\ell - t$, then $f(x, y) = (a - b, a + b, X^{-1}(t))$. If for integers $a, b \geq 0$ and $t \in \{x_1, \dots, x_{|T|}\}$ it is the case that $x = 2wa + w$ and $y = b\ell + \ell/2 - t$, then $f(x, y) = (a - b, a + b + 1, X^{-1}(t))$. If (x, y) does not satisfy either of these constraints, then $f(x, y)$ is undefined.

Given the partial mapping f , for a slider assembly A we say A maps to assembly A' over T if A' is the assembly obtained by including each tile of type t at position (w, y) such that $f(x, y) = (w, u, t)$ for some slider in A at position (x, y) . If any slider in A is at a position at which f is not defined, then A does not have a defined mapping to a square aTAM tile assembly over T .

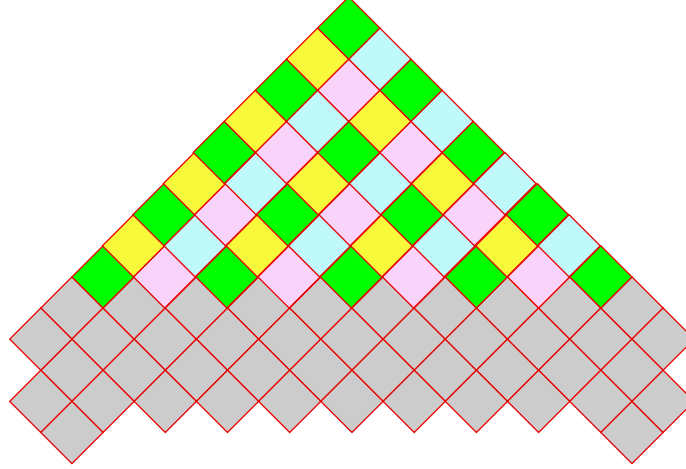


Figure 12: Fat-seeded, double-checkerboarded pyramid systems start with a seed assembly shown in grey and grow upwards with cooperative temperature-2 bonding, yielding a maximum possible assembly in the shape of a pyramid. Additionally, the system is double-checkerboarded, meaning a placed tile is distinct from its north, south, east, west, northwest, southeast, northeast, and southwest neighbors.

Simulating an aTAM system with Sliders We say a slider system $v = (SLDR_\gamma^X, 3, s')$ *terminally simulates* an aTAM square-tile system $\gamma = (T, \tau, s)$ if the set of terminal assemblies $TERM_v$ maps exactly to the set $TERM_\gamma$ when the mapping of slider assemblies is applied to each element of $TERM_v$.² For the remainder of this section we simply use the term *simulates* to refer to terminal simulations.

Pyramid aTAM Systems An aTAM system $\Gamma = (T, 2, s)$ is said to be a *fat-seed pyramid system* if 1) s contains some number n of tiles configured in the format described in Figure 12, with the property that all adjacent tile edges match glues, and 2) all glues in T have strength 1, and 3) the tile set T and seed s are such that no tiles can attach to the southern face of the seed. In addition, a fat-seed pyramid system is said to be *double-checkerboarded* if for any attachable tile during the assembly process, the attached tile, its southwest, southeast, and southern neighbors are all distinct tile types. An example of a coloring scheme that denotes which tiles must be of differing type is shown in Figure 12.

Planar Assembly A pftTAM system (T, τ, s) is said to be *planar* if for each possible tile attachment for all assembly sequences, the attached tile is guaranteed to have a collision-free path within the plane to slide into attachment position. In general, planar assembly systems are desirable in that they offer the possibility for implementation within a system that is restricted to assembly on a surface, and further inform what constructions might generalize into 3 dimensions.

²This is a weaker definition of simulation than what is considered in [6, 3] in that it does not model equivalent dynamics. While our construction actually satisfies a stronger definition of simulation, we omit the more involved simulation definition for simplicity.

6.1.2 Simulation of Cellular Automata

In this section we show that for any double-checkerboarded, fat-seed pyramid aTAM system Γ , the single-tile, translation-only slider tile system derived from Γ simulates Γ . It has been shown that fat-seed pyramid systems with size n seeds are capable of simulating the first $O(n)$ steps of 1D blocked cellular automata [23, 13], a universal class of cellular automata. Thus, our result yields a single-tile translation-only system for simulating universal computation.

We first establish a lemma that makes an algebraic claim about sequences of positive integers. This allows us to assign appropriately spaced glues to our single slider tile, such that certain undesired alignments are infeasible, and further that this can be done with a small polynomially-sized slider tile.

Lemma 6.1 *There is a set $\{x_1, x_2, \dots, x_k\}$ of (distinct) integers in the range $[1, 3k^5]$ such that, for any indices a, b, c, d, e , and f , we have $x_a + x_b + x_c = x_d + x_e + x_f$ if and only if the equation holds algebraically, i.e., $\{a, b, c\} = \{d, e, f\}$.*

Proof: We set the x_i 's incrementally, mimicking a construction in the fusion tree data structure of Fredman and Willard [7]. First we set $x_1 = 1$. If we have so far set x_1, x_2, \dots, x_{i-1} , we set x_i as follows. For any indices $a, b, c, d, e, f \in \{1, 2, \dots, i-1\}$ for which $x_a + x_b + x_c = x_d + x_e + x_f$ does not hold algebraically, the new equations involving one, two, or three copies of x_i —

$$\begin{aligned} x_i + x_b + x_c &= x_d + x_e + x_f, \\ x_i + x_i + x_c &= x_d + x_e + x_f, \\ x_i + x_i + x_i &= x_d + x_e + x_f \end{aligned}$$

—have unique solutions for x_i :

$$\begin{aligned} x_i &= -x_b - x_c + x_d + x_e + x_f, \\ x_i &= \frac{1}{2}(-x_c + x_d + x_e + x_f), \\ x_i &= \frac{1}{3}(x_d + x_e + x_f). \end{aligned}$$

Thus, if we set x_i to avoid these $\leq 3(i-1)^5$ bad values, then we guarantee the theorem holds on $\{x_1, x_2, \dots, x_i\}$. (In particular, $x_i + x_b + x_c = x_j + x_b + x_c$ will hold only if $i = j$, so x_i is distinct from previously chosen x_j 's.) Setting x_i is possible provided the number of choices for x_i is greater than the number of bad values, i.e., $3k^5 > 3(i-1)^5$, which follows from $i \leq k$. Once we finally set x_k , we have the desired set $\{x_1, x_2, \dots, x_k\}$. \square

We now leverage our slider construction and the above properties for the main result of this subsection.

Theorem 6.2 *For any double-checkerboarded, fat-seeded pyramid aTAM system $\Gamma = (T, 2, s)$, there exists a single-tile, translation-only pfbTAM system that simulates Γ . Further, the single tile of the simulating system is of size $O(|T|^5)$, and the system satisfies the planar assembly constraint.*

Proof: Consider an arbitrary double-checkerboarded, fat-seed pyramid aTAM system $\Gamma = (T, 2, s)$ with size n seed. We will prove the theorem by showing that the single-tile translation-only system $\beta = (SLDR_\Gamma^X, 3, s')$ is a planar simulation of Γ , where $SLDR_\Gamma^X$ is a slider derived from Γ according

to Section 6.1.1, and s' is the assembly over $SLDR_{\Gamma}^X$ that maps to s according to the assembly mapping described in Section 6.1.1.

First, we show that $\beta = (SLDR_{\Gamma}^X, 3, s')$ satisfies the requirement that s' is stable. Note that any cut of a fat-seed must separate at least 3 neighbor tiles, with a “neighbor” tile being any tile directly north, south, northeast, southeast, northwest, or southwest. Therefore, since all adjacent edges of s are assumed to be matching strength-1 glues by the definition of fat-pyramid systems, we know that the seed s' has minimum cut strength of at least 3 (see Figure 12 for a picture example of s').

To finish the argument for correct simulation, we need to show that the set of producible assemblies are the same when the slider assemblies are mapped according to Section 6.1.1. To show this, assume a slider assembly A correctly maps to producible pyramid assembly A' . For any attachable aTAM tile to A' , it is easy to see that there is a corresponding position at which a slider tile can attach to A to obtain a slider assembly that maps to the new aTAM assembly. The crux of the correctness argument lies in showing that any slider that may attach must attach such that the resultant assembly is defined. If the assembly is defined, it is straightforward to verify that the resultant assembly maps to a corresponding producible pyramid assembly. Therefore, we will argue that the attachment of a single slider will maintain that the assembly has a defined mapping to a producible pyramid assembly.

To argue this, we first rule out a number of potential issues. First, an attaching slider must do so by matching each of its southwest, south, and southeast pads with one glue. This holds because all glues have strength 1, and by Lemma 6.1 the sequence of x_i 's are such that at most one pair per pad can line up, assuming a non-perfect alignment. In the case of perfect alignment of a southwest or southeast pad, there will be 0-strength bonding because of the double-checkerboarded structure of the simulated tile system. Finally, perfect alignment of the southern pad is prevented by geometric hindrance in the form of the black bump protrusions at the base of the southern and northern pads.

Therefore, the only way to extend an existing arrangement of tiles is to attach a new tile to three different tiles, making use of the southeast, south, and southwest pads, each with glue strength one. We will proceed by induction to argue that such a newly placed tile will have to be appropriately placed on the underlying grid, with a vertical shift corresponding to the appropriate tile type of the simulated tile system.

More precisely, consider a tile D that is bonded with the existing tiles A, B, C , as indicated in Figure 13. In the following, we discuss the y -distance of the involved slider pegs from the baseline of their respective pads, which causes a vertical deviation from the respective grid positions. In the following, we consider these deviations modulo m , and simply refer to these as “relative positions”. Reference to slider name is indicated by letters a, b, c, d , while indices sw, s, se, ne, n , and nw encode the pegs in directions southwest, south, southeast, northeast, north, northwest; for example, the bond between A and D (shown in green in the figure) is shifted by a (vertical) y -distance of d_s from the baseline of D 's northern bonding pad, and a_n from the baseline of A 's southern bonding pad. By assumption, A, B, C are correctly placed, so that A bonds with B and C (indicated by blue color in the figure) at the peg distance that encodes the tile type of A , i.e., $a_{nw} = a_{ne}$. We denote by $\Delta_y(X, Z)$ the difference of relative positions between two tiles X and Z . Then we have $\Delta_y(B, A) = a_{nw} - b_{se}$ and $\Delta_y(C, A) = a_{ne} - c_{sw}$.

Now assume that there is a bond between D and B (shown in red) that uses pegs at relative positions d_{sw} and a_{ne} , a bond between D and A (shown in green) that uses pegs at relative positions

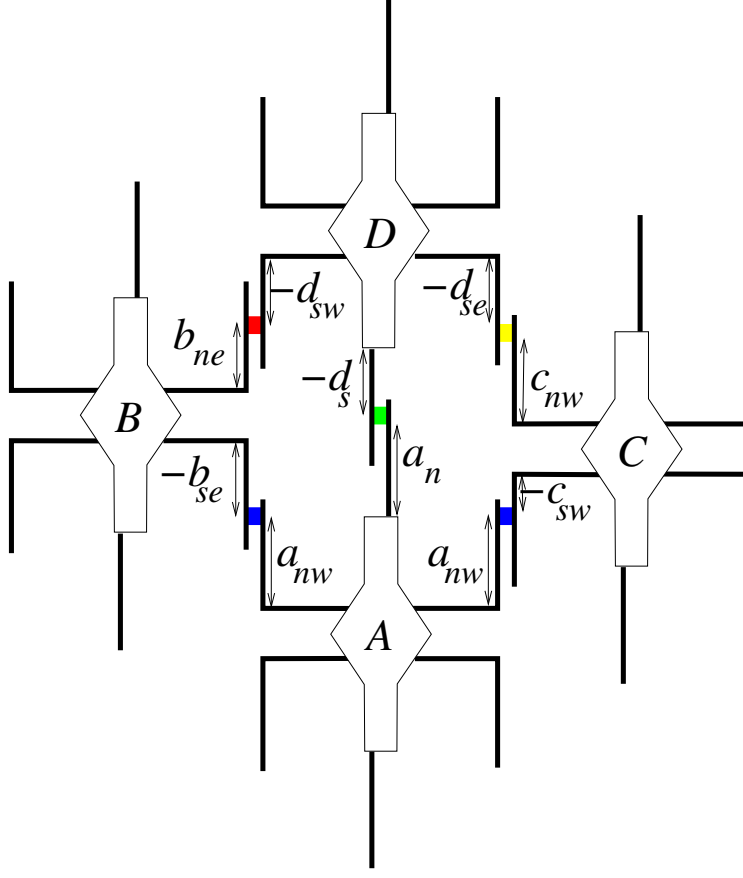


Figure 13: Correctness of the slider simulation: A new slider D is attached to existing sliders, A , B , C , making use of three different bonds, shown in red, green, yellow. Arguing the relationship between the involved y -distances (which are measured from the baseline of each pad, and considered modulo m) shows that this is possible iff a position for D is used that corresponds to an encoding of the corresponding tile in the simulated tile system.

d_s and a_n , and a bond between D and C (shown in yellow) that uses pegs at relative positions d_{se} and c_{nw} . We will argue that this implies that D is correctly placed, with all bonds of any tile using a proper peg position, thus D and its bonds encode a tile of appropriate tile type in the original tile system.

First of all, we observe that $\Delta(D, B) = b_{ne} - d_{sw}$, while $\Delta(D, C) = c_{nw} - d_{se}$, and $\Delta_x(D, A) = a_n - d_s$. Comparing the total shift between the three paths (A, B, D) , (A, C, D) , (A, D) , as shown in the figure, we conclude $\Delta_x(D, B) + \Delta_x(B, A) = \Delta_x(D, C) + \Delta_x(C, A)$, as well as $\Delta_x(D, C) + \Delta_x(C, A) = \Delta_x(D, A)$. From the first equation, we conclude $b_{ne} - d_{sw} + a_{nw} - b_{se} = c_{nw} - d_{se} + a_{ne} - c_{sw}$, so $a_{nw} = a_{ne}$ implies

$$b_{ne} + c_{sw} + d_{se} = b_{se} + c_{nw} + d_{sw}. \quad (1)$$

From the second equation, we conclude

$$a_{nw} + b_{ne} + d_s = a_n + b_{se} + d_{sw}. \quad (2)$$

Now both $b_{ne} = d_{sw}$ and $d_{se} = c_{nw}$ can be excluded, as they would imply a perfect alignment between B and D , or D and C , respectively.

Next, consider $b_{ne} \neq b_{se}$. Then Lemma 6.1 applied to (1) and (2) implies that $d_{se} = b_{se}$, $c_{sw} = d_{sw}$, $b_{ne} = c_{nw}$, $a_{nw} = d_{sw}$, $b_{ne} = a_n$, $d_s = b_{se}$. This implies $c_{sw} = d_{sw} = a_{nw}$; because of $a_{nw} = a_{ne}$, it follows that $a_{ne} = c_{sw}$, i.e., perfect alignment between A and C , which is impossible.

Therefore, we conclude that $b_{ne} = b_{se}$; then Lemma 6.1 applied to (1) implies that $c_{sw} = c_{nw}$, $d_{se} = d_{sw}$, and applied to (2) implies that $a_{sw} = a_n$, $d_s = d_{sw}$, meaning that all bonds of D must use proper pegs for encoding the simulated tiles, as claimed. \square

6.2 Limitations of Translation-Only Systems

The many-gon single-tile self-assembly systems derive their power from being able to rotate. It is natural to ask whether such rotations are necessary for general single-tile simulations, or if it is possible to simulate different tile types by the attachment of a single geometrically complex tile at different relative translations, in an even more general fashion than Subsection 6.1. We show that rotations are necessary in the single-tile model, by proving that translation-only systems with a single tile have very limited power, regardless of the tile's geometric complexity. This holds under rather general assumptions: for the purposes of this section, a tile may be an arbitrary two-dimensional, bounded, connected, regular closed set S , i.e., S is equal to the topological closure of its interior points. In the following, we say that two tiles *overlap*, iff they have non-disjoint interiors; they *touch*, iff they intersect without overlapping. A potential bond between two tiles requires that they touch in more than one point, which must be equipped with a matching glue.

To obtain the impossibility results, we start with a lemma about the translation of connected shapes in 2D. We assume that this lemma has been previously discovered, but we have been unable to find it in the literature.

Lemma 6.3 *Consider a two-dimensional, bounded, connected, regular closed set S , i.e., S is equal to the topological closure of its interior points. Suppose S is translated by a vector v to obtain shape S_v , such that S and S_v do not overlap. Then the shape S_{c*v} obtained by translating S by $c * v$ for any integer $c \neq 0$ also does not overlap S .*

Proof: Assume that there is a smallest integer $c > 1$ for which S and S_{c*v} overlap; we will show that S_v must overlap one of them, implying the claim. Without loss of generality, let $v = (1, 0)$; see Figure 14.

Let x_{\min} be the smallest x -coordinate of S , and let x_{\max} be the largest x -coordinate of S_{c*v} . For a small $\varepsilon > 0$, let $s = (s_x, s_y)$ be an interior point of S with $s_x - x_{\min} < \varepsilon$, and let $t = (t_x, t_y)$ be an interior point of S_{c*v} with $x_{\max} - t_x < \varepsilon$. Because S and S_{c*v} are regular closed, and $S \cup S_{c*v}$ overlap, there is a path P between s and t that stays in the interior of $S \cup S_{c*v}$. Let y_{\max} and y_{\min} be the largest and smallest y -coordinate of points in S . Then we can connect t with s by a vertical line segment ℓ_1 up to y -coordinate y_S , a horizontal line segment ℓ_2 to the x -coordinate of s , and a vertical line segment ℓ_3 to s ; this yields a simple closed curve C .

Because any point of S_v with maximal y -coordinate lies on ℓ_2 , and P lies strictly below ℓ_2 , C must contain an interior point q of S_v in its interior. On the other hand, a point of S_v with y -coordinate y_{\min} must lie below P , and therefore outside of C , so there must be an interior point p of S_v outside of C . Because S_v is connected, there must be a path Q in the interior of S_v that

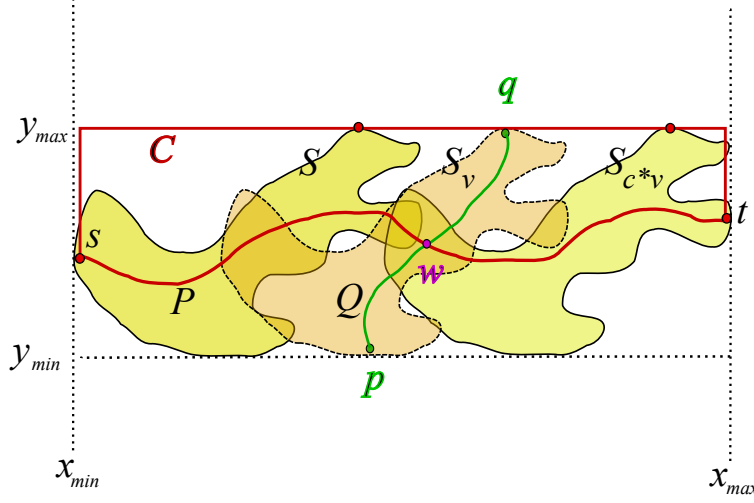


Figure 14: Proof of Lemma 6.3: overlaps between multiple copies of a shape.

connects p and q . Thus, Q must cross C , hence P . Therefore, there is a point w in which S_v and $S \cup S_{c*v}$ overlap, and the claim follows. \square

Theorem 6.4 *For any self-seeding, single-tile, translation-only (non-rotatable, non-flippable) self-assembly system $\Gamma = (T, \tau)$, the set of producible assemblies of Γ is either just the single seed copy of T , or contains assemblies of unbounded size.*

Proof: Consider a seed tile T_0 and suppose that we can attach a translated copy T_1 to T_0 without causing any overlap. Then Lemma 6.3 shows that we can proceed to assemble an unbounded sequence of tile copies T_i , by attaching each T_i to T_{i-1} . \square

Corollary 6.5 *There are aTAM systems that cannot be simulated by a self-seeding, translation-only, 1-tile self-assembly system.*

Corollary 6.5 clearly follows from Theorem 6.4, because there exist singly-seeded aTAM systems (those with seeds consisting of a single tile) with terminal assemblies that have more than one, but a finite number of tiles.

Theorem 6.4 shows an inherent problem with self-seeding translation-only systems: to get started requires a strength- τ attachment between two individual tiles, which leads to infinite growth. This remains true in the more general case of weaker individual bonds, where a tile requires a cooperative bonding between two or more copies of itself to attach in a τ -stable fashion. In particular, for any translation-only single-tile system whose seed is a 3-tile assembly, any additional attachment of even a single tile implies the assembly can grow forever, showing that translation-only single-tile systems are much weaker than rotational systems.

Theorem 6.6 *Let $\Gamma = (T, \tau, S)$ be a self-assembly system with $|T| = 1$, consisting of a single non-rotatable, non-flippable tile T_0 that is closed regular. If Γ 's seed S consists of three copies of T_0 that are bonded in a τ -stable manner, the set of producible assemblies of Γ is either just this seed, or contains assemblies of unbounded size.*

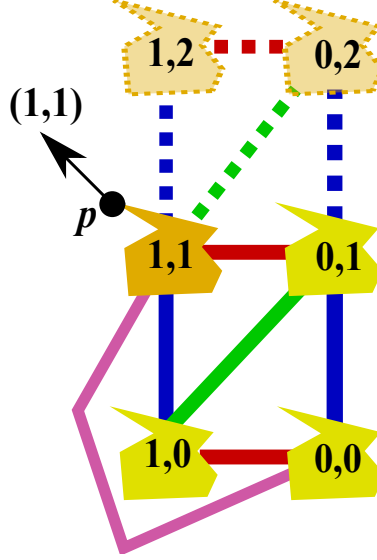


Figure 15: Proof of Theorem 6.6: establishing an unbounded assembly.

Proof: Refer to Figure 15, in which T_0 is shown symbolically by a polygon; note that T_0 's shape may be much more complicated, and that there may be various intersection points between different copies. A tile is in a geometrically feasible position if it does not overlap any existing tile; it can attach to an existing assembly, iff in addition, there are possible bonds of sufficient strength.

We start by considering a seed consisting of three mutually touching copies T_1, T_2, T_3 of T_0 , denoted by the labels $(0,0)$, $(1,0)$, and $(0,1)$ in the figure. Let the respective bonds be denoted by “blue”, “red”, and “green”, with strengths σ_b, σ_r , and σ_g . If the centers of the three tile copies form a collinear arrangement, or if one of the individual bonds has strength at least τ , the claim follows immediately from Lemma 6.3, along the lines of Theorem 6.4. Therefore, we assume that $\sigma_b, \sigma_r, \sigma_g < \tau$, but $\sigma_b + \sigma_r \geq \tau$, $\sigma_b + \sigma_g \geq \tau$, $\sigma_r + \sigma_g \geq \tau$; furthermore, we may assume that the tile centers form a non-degenerate triangle, which spans a two-dimensional vector space described by the basis vectors $(1,0)$ (between T_1 and T_2) and $(0,1)$ (between T_1 and T_3 .) In the following, we will denote tile copies by their respective coordinates, i.e., $T_1 = T(0,0)$, $T_2 = T(1,0)$, $T_3 = T(0,1)$.

Now consider a potential tile $T(1,1)$. Because its position relative to $T(0,1)$ is the same as that of $T(1,0)$ relative to $T(0,0)$, $T(1,1)$ and $T(0,1)$ touch, but their interiors do not overlap, and they can form a red bond. Similarly, $T(1,1)$'s position relative to $T(1,0)$ is the same as that of $T(0,1)$ relative to $T(0,0)$, so $T(1,1)$ and $T(1,0)$ may form a blue bond. Because of $\sigma_b + \sigma_r \geq \tau$, $T(1,1)$ can attach to the seed, provided it is in a geometrically feasible position; because $T(1,1)$ does not overlap $T(0,1)$ or $T(1,0)$, this is the case, iff its interior does not intersect the interior of $T(0,0)$.

So assume that $T(1,1)$ and $T(0,0)$ overlap. (In Figure 15, this is shown symbolically by a purple connection.) Then the intersection graph of $T(0,0)$, $T(1,0)$, $T(0,1)$, $T(1,1)$ is a planar geometric embedding of the complete graph K_4 ; as bonds require more than a single intersection point, we can represent the edge $e_{i,j}$ between any pair of vertices v_i and v_j by a path strictly within the union of the respective tiles, without intersecting any of the two other tiles. Because we have a planar embedding of K_4 , this means that one of the tiles must be strictly inside of a simple closed curve that lies completely within the union of the other three. This leads to a contradiction: an extreme point p of $T(1,1)$ in direction $(1,1)$ cannot be in the convex hull of $T(0,0) \cup T(1,0) \cup T(0,1)$. Similarly,

each other tile has an extreme point outside of the convex hull of the three others. Therefore, the intersection graph of the tiles cannot be a K_4 , showing that $T(1, 1)$ and $T(0, 0)$ cannot overlap.

We conclude that $T(1, 1)$ is a feasible extension of the seed. By Lemma 6.3, $T(0, 0)$ and $T(0, 2)$ cannot overlap; furthermore, we can apply the above reasoning for the non-overlapping position of $T(1, 1)$ and $T(0, 0)$ to show that $T(1, 0)$, $T(0, 1)$, $T(1, 1)$, $T(0, 2)$ cannot form a K_4 , so $T(0, 2)$ and $T(1, 0)$ do not overlap. Hence, we conclude that $T(0, 2)$ is a further feasible extension, using green and blue bonds of combined strength $\sigma_r + \sigma_g \geq \tau$.

This can be iterated: by alternating extensions at $T(1, i)$ and $T(0, i)$, we get an unbounded assembly. \square

Matters get more involved with arbitrary seeds. The following conjecture implies that translation-only, single-tile systems have significantly reduced computing power.

Conjecture 6.7 *Let $\Gamma = (T, \tau)$ be a self-assembly system with $|T| = 1$, consisting of a single non-rotatable, non-flippable tile P that is closed regular, i.e., a closed two-dimensional set that is equal to the closed hull of its interior points. If Γ is self-seeded with a finite number n of copies of P that are bonded in a τ -stable manner, then (regardless of the geometric complexity of P), any producible assembly of Γ consists either of $O(n^2)$ tiles, or is unbounded. Furthermore, the diameter of any finite assembly will be linear in the diameter of the seed.*

A proof should be based on Lemma 6.3, Theorem 6.6, and exploit the planarity of the tile-touching graph during the assembly, in particular the length of its exterior face.

Our slider construction yields an assembly with $\Omega(n^2)$ tiles for a seed with n tiles, so the bound would be tight.

7 Plane Tilings

Here we apply the idea of bump-and-dent geometry in the n -gon polygon tile assembly construction to another, substantially older type of “tile system”. Plane tiling has been studied for many decades, and the standard problem in this area is the following: given a set of polygon-shaped tiles with patterns on them, find an infinite arrangement of these tiles that covers the plane, subject to satisfying constraints on the patterns. Wang tiles are one such family of plane tiling systems: a Wang tile consists of a square tile with each edge colored, with the constraint that each edge must be matched to an adjacent edge of another tile with the same color. However, there are a large number of other plane tiling systems such as Penrose tilings, Robinson tilings, and the interlocking tessellations of M.C. Escher.

The complexity of such systems comes from extending both the set of transformations the tiles may undergo and the rules enforced on adjacent tiles. With the notation of Section 2.4, classical square Wang tiles belong to the plane tiling family $(\{\}, c_m)$, and so may not be rotated or flipped; adjacent tiles must have identical colors on coincident tile sides. Square Robinson tiles [17], on the other hand (belonging to $(\{t_r, t_f\}, c_c)$), may be rotated or flipped, but have complementary color patterns on coincident tile sides (see Figure 16).

Clearly, Wang tile systems have a strong resemblance to the square-tile aTAM systems discussed in this paper. However, the models are qualitatively different: the behavior of a set of Wang tiles, and a similar aTAM system with edge colors exchanged for glues, are unrelated. For instance, coincident edges of adjacent tiles in an *assembly* may have different glues while coincident edges in

a *plane tiling* are required to have matching colors. Also, partially completed tile assemblies are guaranteed to be contained in some terminal assembly (and are computable in polynomial time on a Turing machine), while partial tilings are *not* guaranteed to be contained in any plane tiling.

Nevertheless, the modest similarities are sufficient to adapt the bump-and-dent approach from Section 5 to create a canonical set of nearly-plane tiling systems that simulate any plane tiling system of colored squares or hexagons.

Our results We consider the set of plane tiling systems covered under the definition found in Section 2.4, restricted to squares and hexagons with colored patterns and/or small surface geometries along their edges. Essentially, we provide a straightforward method to convert each member of this wide class of plane tiling systems into a single-tile system on the same lattice. Specifically, we show how any such tile system in the families $(\{\}, c_m)$, $(\{t_r, t_f\}, c_m)$, $(\{\}, c_c)$, and $(\{t_r, t_f\}, c_c)$, can be simulated by a single-tile nearly-plane tiling system in the family $(\{t_r, t_f\}, c_c)$.

Theorem 7.1 *Each colored square and hexagon plane tiling system in the families $(\{\}, c_m)$, $(\{t_r, t_f\}, c_m)$, $(\{\}, c_c)$ and $(\{t_r, t_f\}, c_c)$ is simulated by an n -gon nearly-plane tiling system.*

We show that as a corollary there exist single-tile plane tiling systems that are aperiodic. We also get that there are single-tile plane tiling systems that are intrinsically universal: they simulate *all systems* with permitted sets of transformations or tile-adjacency constraints.

An n -gon nearly-plane tiling system Our simulator single-tile plane tiling systems borrow the idea of bump-and-dent geometry found in the self-assembly construction in Section 5, but modify its usage to fit the setting of plane tilings (as opposed to a self-assembly-based pfbTAM model). This resulting tiling system consists of a single polygon that is a convex regular polygon with small geometry added to each side. We add the special constraint that any valid tiling using this system must consist of tiles placed on a square or hexagonal lattice, with tiles adjacent on the lattice meeting at a pair of sides that have matching geometry (see Section 2 for definitions) and matching colors. Note that this does not completely cover the plane (hence the qualifier “nearly”), but does produce a dense infinite pattern.

7.1 Simulating plane tiling systems

Recall that in Section 5, unwanted rotations of tiles are eliminated by replacing glues on each pair of opposing sides with colors (glues) unique to that direction pair (via a numbering scheme for creating glue subsets). The remaining unwanted rotations are eliminated via geometry, using a bump-and-dent pair located on the clockwise or counterclockwise end of each edge. The result was that a hTAM system (similar to a hexagonal plane tiling system of the family $(\{\}, c_m)$) was converted to a single-tile pfbTAM system (similar to a single-tile plan tiling system of the family $(\{t_r, t_f\}, c_m)$). Simulating systems in the family $(\{t_r, t_f\}, c_m)$ is possible by eliminating all direction-specific glues and surface geometry.

Lemma 7.2 *All colored square and hexagonal plane tiling systems in the families $(\{\}, c_m)$ and $(\{t_r, t_f\}, c_m)$ can be simulated by an n -gon nearly-plane tiling system.*

Proof: Restricting the n -gons to live on lattice of S and adjoin only other n -gons with matching geometry and colors (glues in the pfbTAM construction) yields a construction simulating systems with $T = \{\}$. Eliminating direction-specific glues and surface geometry leaves the n -gon tile free to rotate and reflect, thus simulating systems with $T = \{t_r, t_m\}$ directly. \square

Next we consider simulating systems with complementary tile adjacency constraints (i.e. $C = c_c$), which we first show is reducible to a complementary geometry constraint. Given some surface geometry, convert the geometry into a pattern by mapping the height of the geometry relative to the flat geometry to a color value (dents have negative values). The opposing mapping is also possible: for a given pattern consisting of a function from distance along the tile's side to a color, map each color to a height of bump. Division of the colors into primal-dual sets of complementarity yields a partition of the colors into bump colors and dent colors.

Given a translation-only geometry-complementation system, create a distinct color for each opposite-direction pair, use the clockwise-counterclockwise placement of bumps-and-dents as in the construction in Section 5, and place a small copy of the geometry from the side of each tile in a small interval along the corresponding side of the n -gon tile. Given a system in the family $(\{t_r, t_f\}, c_c)$, create a similar construction but with a single 'blank' color on all sides of the n -gon.

Lemma 7.3 *All colored square and hexagon plane tiling systems in the families $(\{\}, c_c)$ and $(\{t_r, t_f\}, c_c)$ can be simulated by an n -gon nearly-plane tiling system.*

Proof: Simulating systems from the family $(\{\}, c_c)$ using the described construction does not allow any rotations except those corresponding to an unrotated and unreflected tile in the original system, as the distinct glue pairs and geometry forbid unwanted orientations for the same reasons as in Section 5. Systems from the family $(\{t_r, t_f\}, c_c)$ are also simulated correctly, as any rotation or reflection of the n -gon corresponds to an orientation of some tile in the simulated set, and has the same geometry as the simulated tile. \square

7.2 An example simulation

A small aperiodic set of square tiles First we describe the ten-tile square plane tile set from the family $(\{t_r, t_f\}, c_c)$ of Robinson [16] that yields only aperiodic tilings of the plane. This system uses a pattern arrows on each tile, with the constraint that every arrow head must meet an arrow tail on an adjacent tile and vice versa: every arrow tail must meet an arrow head. Figure 16 shows the tile set itself with the aperiodic pattern (orange and blue) and the parity-enforcing pattern (grey) and Figure 17 shows an example tiling using this tile set.

The blue and orange arrows represent constraints in the five 'basic tiles' of Robinson Figure 2. The grey arrows represent parity constraints introduced in Robinson Figure 3 to eliminate the corner geometry of the six (l-most) square tile set found in Robinson Figure 1.

The simulating n -gon system We use the construction described previously to convert this ten-tile system into an n -gon nearly-plane tiling system. Figure 18 shows the resulting $4c$ -gon tile.

7.3 One tile that simulates Wang tilings

The previous construction showed that it is possible to take Robinson's plane tiling system, and convert that to a nearly-plane tiling system with a single n -gon tile. We can apply this transformation idea to *any* plane tiling system on the square or hexagonal grid in order to get a single

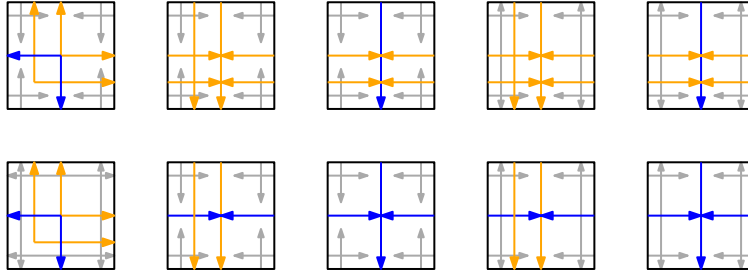


Figure 16: A set of ten aperiodic tiles from Robinson. The pattern constraint requires every arrow head on the surface of a tile to be met by a parallel arrow tail on an adjacent tile and vice versa.

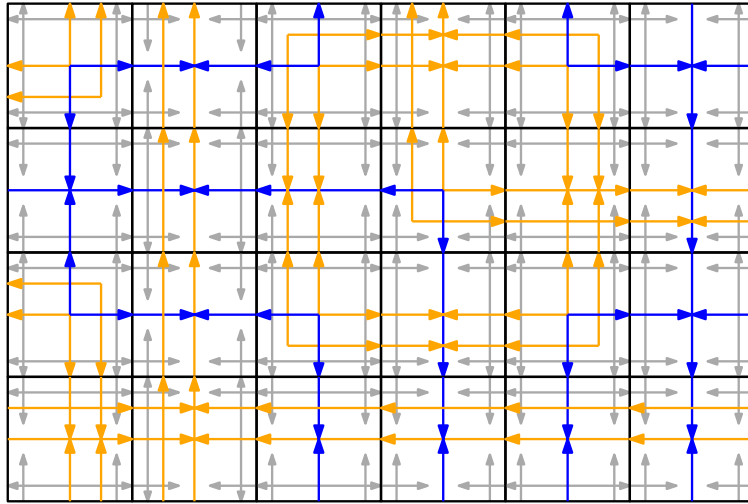


Figure 17: A (partial) tiling of the plane using the tile set from Robinson.

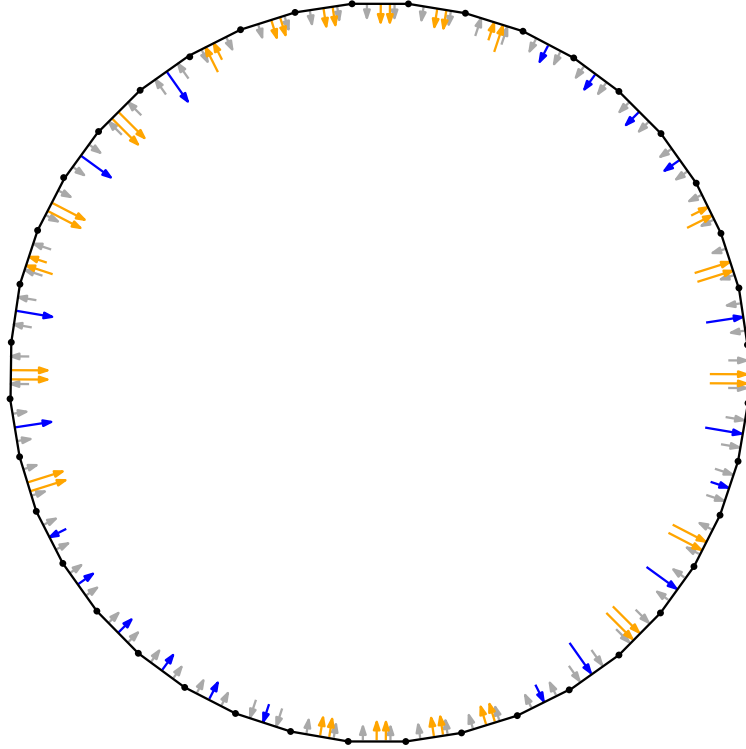


Figure 18: The single tile type generated from Robinson's set of ten square tiles. The sides of the tile correspond to the sides of every tile in the Robinson tile set such that if the tile's uppermost side is aligned horizontally then the four horizontal and vertical edges correspond to an orientation of a tile in the Robinson set.

tile, nearly-plane tiling system. As an immediate corollary, the single tile simulator inherits all (non-trivial) properties of the system being simulated. For example, we know from the previous section that there is a single tile system that tiles the plane aperiodically.

Lafitte and Weiss introduce simulations between tilings [12]. They show the existence of universal tilings: such a tiling is capable of simulating a countably infinite subset of the tilings from *each* Wang tile set: changing the scale factor lets us simulate a new tiling, from a new, or the same, tile set.³ Via our construction, obtain a single tile that is capable of these kinds of simulation.

Acknowledgements

This work was initiated at the 27th Bellairs Winter Workshop on Computational Geometry held on February 11-17, 2012 in Holetown, Barbados. We thank the other participants of that workshop for a fruitful and collaborative environment. In particular, we thank Brad Ballinger and Anna Lubiw for important discussions regarding Lemma 6.3. In addition, we thank Jarkko Kari for interesting and fruitful discussions on aperiodic tilings of the plane with few tile types.

References

- [1] R. Berger. *The Undecidability of the domino problem*. PhD thesis, Harvard University, 1964.
- [2] R. Berger. The undecidability of the domino problem. *Memoirs of the American Mathematical Society*, 66, 1966.
- [3] S. Cannon, E. D. Demaine, M. L. Demaine, S. Eisenstat, M. J. Patitz, R. Schweller, S. M. Summers, and A. Winslow. Two hands are better than one (up to constant factors). Technical Report 1201.1650, Computing Research Repository, 2012.
- [4] K. Culik, II. An aperiodic set of 13 Wang tiles. *Discrete Mathematics*, 160(1-3):245–251, Nov. 1996.
- [5] E. D. Demaine and M. L. Demaine. Jigsaw puzzles, edge matching, and polyomino packing: Connections and complexity. *Graphs and Combinatorics*, 23 (Supplement):195–208, 2007.
- [6] D. Doty, J. H. Lutz, M. J. Patitz, R. T. Schweller, S. M. Summers, and D. Woods. The tile assembly model is intrinsically universal. In *Proceedings of the 53rd Annual IEEE Symposium on Foundations of Computer Science*, pages 302–310, October 2012.
- [7] M. L. Fredman and D. E. Willard. Surpassing the information theoretic bound with fusion trees. *Journal of Computer and System Sciences*, 47(3):424–436, 1993.
- [8] B. Fu, M. J. Patitz, R. Schweller, and R. Sheline. Self-assembly with geometric tiles. In *ICALP 2012: Proceedings of the 39th International Colloquium on Automata, Languages and Programming*, pages 714–725, Warwick, UK, 2012.
- [9] B. Grünbaum and G. C. Shephard. *Tilings and patterns*. W. H. Freeman, 1987.

³As they point out, the *countable* restriction is somehow necessary, because having a single tiling that simulates all tilings, is impossible, because there is a countable set of representation functions but an uncountable set of tilings.

- [10] P. Gummelt. Penrose tilings as coverings of congruent decagons. *Geometriae Dedicata*, 62(1):1–17, 1996.
- [11] L. Kari, S. Seki, and Z. Xu. Triangular and hexagonal tile self-assembly systems. In *Proceedings of the 2012 International Conference on Theoretical Computer Science: Computation, Physics and Beyond*, pages 357–375, Auckland, New Zealand, 2012.
- [12] G. Lafitte and M. Weiss. Universal tilings. In W. Thomas and P. Weil, editors, *STACS 2007, 24th Annual Symposium on Theoretical Aspects of Computer Science, Aachen, Germany, February 22-24, 2007, Proceedings*, volume 4393 of *Lecture Notes in Computer Science*, pages 367–380. Springer, 2007.
- [13] N. Margolus. Physics-like models of computation. *Physica D: Nonlinear Phenomena*, 10(1):81–95, 1984.
- [14] T. Neary and D. Woods. The complexity of small universal Turing machines: a survey. In *SOFSEM 2012: Theory and Practice of Computer Science*, pages 385–405. Springer, 2012.
- [15] R. Penrose. Set of tiles for covering a surface. US Patent 4133152, January 1979.
- [16] R. Robinson. Undecidability and nonperiodicity for tilings of the plane. *Inventiones Mathematicae*, 12:177–209, 1971.
- [17] R. M. Robinson. Undecidability and nonperiodicity for tilings of the plane. *Inventiones Mathematicae*, 12(3):177–209, 1971.
- [18] P. W. K. Rothmund. Design of DNA origami. In *ICCAD '05: Proceedings of the 2005 IEEE/ACM International conference on Computer-aided design*, pages 471–478, Washington, DC, USA, 2005. IEEE Computer Society.
- [19] R. Schulman and E. Winfree. Synthesis of crystals with a programmable kinetic barrier to nucleation. *Proceedings of the National Academy of Sciences*, 104(39):15236–15241, 2007.
- [20] R. Schulman and E. Winfree. Programmable control of nucleation for algorithmic self-assembly. *SIAM Journal on Computing*, 39:1581–1616, 2009.
- [21] J. E. S. Socolar and J. M. Taylor. Forcing nonperiodicity with a single tile. *The Mathematical Intelligencer*, 34(1):18–28, Spring 2012.
- [22] H. Wang. Proving theorems by pattern recognition—II. *The Bell System Technical Journal*, 40(1):1–41, January 1961.
- [23] E. Winfree. *Algorithmic Self-Assembly of DNA*. PhD thesis, California Institute of Technology, June 1998.
- [24] S. Woo and P. W. Rothmund. Stacking bonds: Programming molecular recognition based on the geometry of DNA nanostructures. *Nature Chemistry*, 3:620–627, August 2011.